

Open Research Online

The Open University's repository of research publications and other research outputs

Arguement in the humanities: A knowledge based approach

Thesis

How to cite:

Stutt, Arthur (1989). Arguement in the humanities: A knowledge based approach. PhD thesis The Open University.

For guidance on citations see [FAQs](#).

© 1989 The Author



<https://creativecommons.org/licenses/by-nc-nd/4.0/>

Version: Version of Record

Link(s) to article on publisher's website:

<http://dx.doi.org/doi:10.21954/ou.ro.0000dfca>

Copyright and Moral Rights for the articles on this site are retained by the individual authors and/or other copyright owners. For more information on Open Research Online's data [policy](#) on reuse of materials please consult the policies page.

oro.open.ac.uk

DX 86715

UNRESTRICTED

**Argument in the humanities:
a knowledge-based approach**

Arthur Stutt

**Thesis submitted in fulfillment
of the requirements for a
PhD in Cognitive Science**

10th April 1989

**Human Cognition Research Laboratory
The Open University
Milton Keynes
MK7 6AA
U.K.**

Author's number: M7022983

Date of submission: 10th April 1989

Date of award: 23rd June 1989

Abstract

In this thesis I have a threefold purpose. I will attempt: (a) to present a generic design for a tool - the Argument Support Program - which can be of use in supporting the reasoning of archaeologists (and others especially, but not exclusively, in the humanities); (b) I will present a model of argumentation and debate as the theoretical orientation within which the model is developed; and, (c) I will suggest that this approach is a natural development of several strands of research within the artificial intelligence community. A tripartite model of argument is presented in terms of arguers, the argument structure produced and the argument domain or field. This model subsumes reasoning, interpretation and argument exchange or debate. It is maintained, further, that while this model is generally applicable, specific domains have particular *styles* of argument. The notion of argument style is discussed in terms of the types of reasoning used. The related concept of relevance in argument is discussed in terms of the specific tokens of these types which may be used in a particular argument. It is argued that archaeology is characterized, at least in part, by the use of argument by analogy and argument from theoretical principles or models. A design for a generic program - the Argument Support Program (ASP) - based on the theoretical principles is delineated. Details of the partial implementation of the model as a constrained debater in the domain of archaeology (ASP for archaeology or ASParch) are presented. Example runs which illustrate how the characterizing features of archaeology are dealt with are also presented as are examples of the various domain and system knowledge bases needed. The application of ASPs to other domains and areas such as literary criticism, legal reasoning and Darwinian theory is discussed. In the final chapter, the achievements and inadequacies of this research are summarized, possible reasons are presented for the inadequacies in the resulting system and future directions discussed.

For Fiona

To be honest I put myself in the category of people who are best able to give form to their ideas by arguing - I entirely subscribe to the view that truth is reached through dispute. Left to study a question on my own, I tend to fall into a reflective state which suits the metaphysical bent of my character and is not conducive to an energetic, creative thought process, since it affords only emotional material with which to construct a - more or less well-ordered - framework for my ideas.

Andrey Tarkovsky - Sculpting in Time

All testing, all confirmation and disconfirmation of a hypothesis takes place already within a system. And this system is not a more or less arbitrary and doubtful point of departure for all our arguments: no, it belongs to the essence of what we call an argument. The system is not so much the point of departure, as the element in which arguments have their life.

Wittgenstein - On Certainty

Acknowledgements

I want to thank everyone at HCRL for a stimulating and comfortable environment and for the many opportunities afforded me to discuss the contents of this thesis and to practice my own skills in argumentation.

I also want to thank the archaeologists who have listened to my ideas and encouraged me to complete the project.

Finally, I want to thank my parents for giving me the opportunity to set out on my present capricious journey.

Contents

Preface	i
List of Figures	ii
 Chapter 1 - Setting the scene	 1
1.1 The application of artificial intelligence techniques to argumentation	1
1.2 The domain of archaeology	5
1.3 Objections to AI	8
1.4 The thesis - model, program design, program	11
1.5 Examples	15
1.6 Conclusion	20
1.7 Guide to the reader	21
 Chapter 2 -Archaeological reasoning	 23
2.1 Current work in using AI programs in archaeology	23
2.2 Archaeological reasoning and analogy	29
2.3 The archaeological domains	36
2.4 An archaeological argument	38
2.5 Conclusion	38
 Chapter 3 - A review of related AI work	 39
3.1 Introduction	39
3.2 Approaches to explanation	47
3.3 Approaches to argumentation	57
3.4 Conclusion	72
 Chapter 4 - Models of interpretation and argument	 74
4.1 Interpretation	74
4.2 Argumentation	82
4.3 Argument style	95
4.4 Relevance	106
4.5 Conclusion	109
 Chapter 5 - Design principles and design	 110
5.1 Preamble about models in AI	110
5.2 Design principles	112

5.3 Some definitions	115
5.4 The architecture of the argument system	120
5.5 The architecture of the interpreter	140
5.6 Conclusion	144
 Chapter 6 - Implementation and examples	 145
6.1 The implementation of ASParch	146
6.2 The examples	162
6.3 Future extensions to ASParch	174
6.4 Conclusion	181
 Chapter 7 - Other argument fields	 182
7.1 Literature	184
7.2 Legal reasoning	188
7.3 Darwinian theory	191
7.4 Conclusion	193
 Chapter 8 - Discussion	 196
8.1 What has been achieved?	196
8.2 Weaknesses in the approach and some possible reasons for these	202
8.3 Future directions	217
 References	 223
 Appendix I - Selections from the Prolog code for ASParch	 241
Appendix II - The Early Bronze Age 'Wessex' culture	267
Appendix III - Macintosh™ screen snapshots for example 1(a) in chapter 6	270
Appendix IV - Results of assessment algorithm on examples 1-3	271

Preface

Parts of chapters 1, 2, 3, 4 and 5 are based on material from Stutt, 1987, 1988a, forthcoming. These reports and papers summarize aspects of the research discussed in this thesis and have been extensively reworked for inclusion here. Chapter 5 includes material from Stutt, 1988b. Sections 4.1 and 5.5 include material based on Patel and Stutt, 1988, 1989. These sections summarize parts of some joint research conducted by myself and my colleague Jitu Patel on the design and implementation of the KIVA archaeological interpreter.

Since a thesis on argumentation imposes on its author the obligation to be reasonably clear as to what arguments are being advanced I have adopted the practice of summarizing the main points made in each chapter in a prefatory *argument*. This practice was once common-place among writers of fiction and non-fiction. It seems appropriate to renew it here, to the benefit, I hope, of the reader.

List of figures and tables

<i>Figure number</i>	<i>Figure title</i>	<i>After page</i>
1.1	The three levels of argument	11
1.2	The architecture of an Argument Support Program	14
1.3	The structure of a hypothetical argument	16
1.4	A hypothetical argument	18
1.5	A response to the hypothetical argument	18
1.6	An ASP generated argument	18
2.1	A logicist analysis	25
2.2	An idealized Pueblo site	36
2.3	An archaeological argument about Durrington Walls	38
2.4	An analysis of the archaeological argument	38
3.1	VanLehn's argument matrix	68
3.2	Marshall's graphical Toulmin structure	69
4.1	Archaeological interpretation	76
4.2	A full toulmin structure	85
4.3	A simple toulmin structure	86
4.4	The tripartite model of argumentation	89
4.5	An argument-1	91
4.6	Argument-2s and topics	94
4.7	Argument-1s and contexts	94
4.8	Debate about <i>The Last Temptation</i> and its background	105
5.1	The program design	110
5.2	The graphical argument-0 or toulmin structure	136
5.3	The graphical argument-1 or interpretation	136
5.4	The hierarchies involved in archaeological interpretation	141
6.1 - 6.10	Example 1(a) - Macintosh™ output screens	164
6.11	Example 1(a) - redrawn screens	165
6.12	Example 1(b) - redrawn screens	167
6.13	Example 2 - redrawn screens	168
6.14	Example 3 - redrawn screens	171
6.15	Finding a weak principle	173

7.1	Levels of reasoning in archaeology	184
7.2	Olsen's diagram of an interpretation of Hamlet	186
7.3	A possible exchange over Olsen's interpretation	187
7.4	A legal argument	190
7.5	An argument from Darwin	192
7.6	Dating the earth	193
8.1	The hybrid system	217
A.1	Example 1(a) system interpretation 1	270
A.2	Example 1(a) user interpretation 1	270
A.3	Example 1(a) system interpretation	270
A.4	Example 1(a) user interpretation 2	270
<i>Table number</i>	<i>Table title</i>	<i>Page</i>
4.1	Formal and informal examples of the three levels of argument	90
4.2	Operations and data structures at each level	92
4.3	A range of fields and their concomitant argument styles	98
5.1	Data structures and operations for an ASP	121
8.1	Possible uses for the successor to ASParch	219
A.1	The results of the assessment algorithm	271

Chapter 1 - Setting the scene

Argument: Argumentation is ubiquitous and innate. It has general and domain dependent aspects. It can be beneficially represented in a computer program. This can be illustrated using the domain of archaeology. The objections of potential users can be overcome by the use of such a program.

1.1 The application of artificial intelligence techniques to argumentation

'Man was born free, and is everywhere in chains of argument.' This pastiche of Rousseau's famous words from the *Social Contract* serves to remind us of the ubiquity of argument and debate. There are arguments everywhere in the media. Ministers defend aspects of government policy in chat shows, intellectuals debate the great moral questions in the early hours, newspaper editorials discuss matters of the day. We engage in argument with our colleagues, our friends and of course our spouses. Since argument and debate are everywhere, to seek an understanding of their nature and function must be a worthwhile enterprise. There is nothing new in the attempt to answer such questions. Logicians, rhetoricians and philosophers have long considered aspects of argumentation. The present thesis looks at argumentation from the point of view of the discipline of artificial intelligence (AI). Thus the emphasis is on the representation of arguments and the knowledge of how to argue rather than on, say, the nature of deduction or the use of certain verbal techniques to persuade. The attempt is made to construct a model of argumentation and debate which could feasibly be tested on a computer. This forces us to construct a model which is more rigorous than natural language accounts but which avoids the normative strictures of philosophical logicians. The model draws upon both *a priori* discussions of the nature of argument and actually occurring arguments. This combination of rigour and realism serves both as an illustration of the acceptable use of artificial intelligence and as a means of clarifying certain aspects of everyday and academic arguing.

As we shall see in chapter 3, this is by no means the first attempt to deal with argumentation using artificial intelligence techniques. The present work differs in that, as well as producing a model of argumentation and debate, it (a) shows how this model could be used to produce a tool to support arguers from the humanities and (b)

shows how the model can be used to solve some of the problems posed by the need to create viable user/expert system interfaces. Because of (a) it also differs from most previous research in argumentation in artificial intelligence by concentrating on a single domain.

The model depends on the fact that argumentation and debate are broadly the same in whatever domain they are used. If this were not so, it would be difficult to build a model which had more than passing application to more than one domain. Consider the following. It's a quotation from *Huckleberry Finn* (Twain, 1987). Huck and Jim are discussing the question of why the French don't speak English:

'... That's a Frenchman's way of saying it.'

'Well, it's a blame' ridicklous way, en I doan' want to hear no mo' about it. Dey ain' no sense in it.'

'Looky here, Jim; does a cat talk like we do?'

'No, a cat don't.'

'Well, does a cow?'

'No, a cow don't, nuther.'

'Does a cat talk like a cow, or a cow talk like a cat?'

'No, dey don't.'

'It's natural and right for 'em to talk different from each other, ain't it?'

' 'Course.'

'And ain't it natural and right for a cat and a cow to talk different from *us*?'

'Why, mos' sholy it is.'

'Well, then, why ain't it natural and right for a *Frenchman* to talk different from us?'

You answer me that.'

'Is a cat a man, Huck?'

'No.'

'Well, den, dey ain't no sense in a cat talkin' like a man. Is a cow a man? - er is a cow a cat?'

'No, she ain't either of them.'

'Well, den, she ain' got no business to talk like either one er the yuther of 'em. Is a Frenchman a man?'

'Yes.'

'Well, den! Dad blame it, why doan' he *talk* like a man? You answer me *dat*!'

I see it warn't no use wasting words - you can't learn a nigger to argue. So I quit.

Arguments or informal debates have certain general characteristics which are illustrated by the above. They are conducted in natural language, within dialogue exchanges, are elliptical, use non-deductive inferences, are grounded in common

sense views of the world, engage the attention of the participants, refer to whole value systems. In fact the above dialogue exchange is conducted using two different varieties of natural language, two different dialects. The exchange is elliptical in the sense that not all the steps in the argument are mentioned. For example the move from the facts about cows and cats to what is 'natural and right' for cows and cats conceals a chain of abstruse argumentation. Huck and Jim are not disinterested. They are engaged and stimulated by the exchange. It matters to them, even if only for a moment, and Huck is annoyed when he is refuted by Jim. There is the question of values and the related issue of what is common sense to the participants. In fact Huck and Jim live in common sense worlds which only partially overlap. For Jim, but not for Huck, it is plain common sense that Frenchmen should speak like Americans. Huck, as revealed by the penultimate sentence, is himself under thrall to a particular view of the world in which other men can be valued at eight hundred dollars. A view not shared by the author, of course.

The comic irony of that penultimate sentence also expresses a truth about all of us. We all know how to argue. We don't have to be taught how to argue any more than we have to be taught how to talk. It is a natural consequence of our competitive natures and the rich semantic complexity of our language that argument is both necessary and possible. Of course we provide training for specialized forms of argument. For philosophers, say, and lawyers. But the arguments they provide, while perhaps more formal in their expression and more profound in their importance nonetheless depend on the tracing or creation of a web of semantic as well as logical interconnections between the concepts which make up a particular context of discourse. Take the following (semi-formal) argument from Searle (1987):

Axiom 1. Brains cause minds.

Axiom 2. Syntax is not sufficient for semantics.

Axiom 3. Minds have contents; specifically, they have intentional or semantic contents.

Axiom 4. Programs are defined purely formally, or syntactically.

Conclusion 1. Instantiating a program by itself is never sufficient for having a mind (by Axioms 2, 3 and 4).

Conclusion 2. The way the brain functions to cause minds cannot be solely by instantiating a program (Axiom 1 and Conclusion 1).

Conclusion 3. Any artefact that had a mind would have to have causal powers (at least) equivalent to those of the brain (by Axiom 1, trivially).

Conclusion 4. For any artefact that had a mind, the program by itself would not be sufficient for having a mind. The artefact would have to have causal powers equivalent to the brain (by Conclusions 1 and 3).

I would contend that neither Huck nor Jim would have much difficulty in following this argument once certain technical terms such as 'axiom' had been explained. The relations which make up a chain of argumentation are essentially the same in both everyday arguing and in great academic debates. Searle's argument may be expressed in a precise manner but it depends essentially on our innate capacity to unravel skeins of interconnection rather than on the application of a body of formal theory such as the predicate calculus.

On the other hand Huck and Jim might not have seen the point in the argument and certainly would not have been able to produce it. They are in the position of the slave boy in Plato's *Meno* (1964) by means of whom Socrates seeks to establish that we have an innate capacity to understand mathematics. Thus, while the model is broadly applicable to many domains, nonetheless there are differences in the sorts of arguments which arguers produce within different domains. These include the differences in vocabulary and the point in the argument highlighted above. There are also differences in the sorts of inference used and the kinds of argument commonly appealed to. More importantly there are differences (as mentioned above) in the sort of knowledge made use of by different arguers in different domains, the common sense or background understanding which is tacitly appealed to, the different value systems and the particular context of an argument. For instance, the context of the Searle argument includes many items of knowledge which would simply not have been available to mid 19th century working class citizens of Missouri.

Thus, we cannot argue without having some knowledge about the topic we are arguing about (though as the example of Huck and Jim given above shows this may be fairly minimal). Given this, the approach adopted in this thesis incorporates various kinds of knowledge. There are knowledge bases for the domain of archaeology as well as a knowledge base which contains knowledge of how to argue. It is in this sense that the model which will be discussed in the following chapters represents a knowledge-based approach. As well as this, the argumentation model provides for the fact that in arguing we often acquire knowledge. Thus argumentation depends on knowledge and is a source of knowledge.

While we can identify the various characteristics of arguments and debates this thesis will only deal with some of them. This is partly for reasons of scope. A proper analysis and simulation of the various characteristics of argument would be many volumes long. Furthermore, some aspects of argument, for example, the relatedness to value systems are problems which artificial intelligence may never be able to deal with. Thus, the thesis is not about dialogue as such. Of course, as the example of Huck and Jim shows, arguing is a form of dialogue. While this is taken account of in the model to be presented below, the reader expecting to find a discussion of many of the issues normally dealt with under that rubric will be disappointed. Nor does the thesis attempt to deal with natural language issues. The generation and parsing of natural language sentences are both research projects in themselves. It is sufficient for the purposes of the present research that we are always aware that our approach must be compatible with the parsers and generators used elsewhere. Furthermore the thesis does not deal with logical issues. That is, if we mean by logical issues, those to do with formal deductive systems. If we mean the rules which govern at least a large part of our everyday discourses, a descriptive rather than normative discipline, then the present research is to do with logic. On the other hand, the thesis does deal with the means of analysing and representing arguments; of generating arguments using domain knowledge and knowledge of how to argue; of generating responses to arguments; of manipulating the domain knowledge using both deduction and analogical reasoning; of making use of higher level theoretical knowledge.

1.2 The domain of archaeology

Apart from my own interest in these areas there are several good reasons for the use of archaeology as a domain for study. In one sense of course it does not matter which domain I use since the work is applicable to many domains as I will argue in chapter 7. What does matter is that the designer of a knowledge based system has experience of the domain or close and prolonged contact with someone who has such experience. For this reason I have had to concentrate on the one domain. One which I know something about and have access to those more knowledgeable.

Before outlining the reasons for using archaeology I must begin by saying something about archaeology itself (see Chapter 2 for a fuller exposition of the nature of archaeological debate). Archaeology is a wide and varied discipline. Its subject matter ranges from the study of Neolithic earthen monuments in Southern England to

medieval fortifications in Normandy. I shall draw my examples from many diverse areas of archaeology although I shall concentrate on prehistory; that is, the study of periods before the uses of written texts. In this country the delimiting date is usually taken as the Roman Invasion of AD 43. Indiana Jones to the contrary, archaeologists are not concerned with the discovery of long hidden exotic treasures, they are more concerned with the precise recording of many thousands of finds in the hope of coming to understand some small part of the culture of those who used these artifacts. It is also important to note that archaeology is not a totally homogeneous discipline. What discipline is? There are two broad categories and several other ways of classification. The broad categories are into theoretical and practical archaeology. Typically the former cogitates in a nice warm study about the discoveries made by the latter on some wind-swept Welsh hillside with pressures from developers, the weather and recalcitrant staff. Often however the theoretical and practical caps are worn by the same person. The division into applied and theoretical is not as clear as in, for example, physics. Archaeologists can also be classified depending on whether they study a particular culture (e.g. Incas), a particular period (The Bronze Age), a particular form of technology (Steam Engines) or use particular techniques (tree ring dating). This heterogeneity is important since it is one of my claims that different areas have different canons of argument. Thus what I say about archaeology will not hold completely for the law. This distinction can also be made within a discipline. My general conclusion is that while there are certainly areas where the canon is markedly different there is a remarkable degree of overlap. Thus while the lawyer has a different style of argument in defending a client compared with that of the archaeologist in interpreting a site they both make use of many common patterns of argument. This will also be the case within archaeology. However there will be disputes within a sub-domain which are not even intelligible to related domains.

My reasons for using archaeology as my primary domain can be summarized as follows:

Negative reasons

- Work on expert systems has concentrated on domains such as medicine and electronic fault finding. However, it may be the case that much of the problem solving in these domains can be done using algorithms which reason from first principles. That is to say that expert systems (seen as composed of heuristic

knowledge) are no longer needed.

- Very few AI researchers have backgrounds in arts subjects apart from philosophy. This has meant that there has been a tendency not to apply AI techniques in these sorts of area. This is a pity since such subjects are a vital part of any culture. Furthermore, if AI techniques are to be tested rigorously they should be applied in many different domains.

Positive reasons

- The area of the humanities is one in which argument plays a large role. Papers written by humanities specialists are not much concerned with empirical fact or experiment but with theory and argument. A humanities domain thus provides a highly suitable domain in which to test the ideas about argument.
- If AI could get a model of the processes we use in interpreting a site or in evaluating a piece of literature it would have gone a long way towards understanding how we work in general. If it fails then this will show a limitation of AI.
- Domains such as archaeology provide important information which fills out the information we have from disciplines such as psychology about how humans work. Archaeology in particular has much to say about the relation of man to material culture and ritual activity.
- In common with other humanities, archaeology is a self-conscious discipline. Archaeologists argue about the theories they apply. No diagnostic engineer will argue in this way due to the relative fixity of the theoretical background. This is important in two ways:
 - a) it provides a test bed for ideas about argument
 - b) it brings out background ideas and the relationships of these to operational principles.
- Archaeology (and literary criticism) are conducted within various theoretical perspectives such as postmodernism, structuralism and marxism. These

perspectives inform the work of many other human and social scientists. AI must have a proper response or means of accommodating such theories if only because they are so prevalent in work in these areas. At a deeper level they must surely contribute to the conceptual framework that these workers employ in interpreting reality. The study of these processes in archaeology (and literary criticism) will have much to tell AI. Albeit this thesis will only expose the tip of a very large iceberg.

1.3 Objections to AI

In the preceding section I gave various reasons for the use of archaeology as a domain of application for my theoretical perspective on arguing. One point not made there was that archaeologists are fairly resistant to the use of expert systems in their discipline. After a period in which expert systems were greeted as possibly epoch-making, a reaction followed in which the inadequacies in the models contained within expert systems were highlighted. The introduction of argument support programs can be seen as a response to this reaction. I feel that it is necessary for the designer of knowledge based systems to have a clear understanding both of the domain and of the problems in using advanced computer systems as perceived by users in the domain. For this reason I have concentrated on the single domain of archaeology in an attempt to lay the fears of archaeologists to rest and in the hope that the case of archaeology can stand as an emblem for the widespread hostility that expert systems provoke in some while they are welcomed with open arms by others. It is often thought that the hostility can be overcome simply by adding on more and more features. I doubt this very much. It is not simply that the programs are felt to be inadequate (though they are) nor that they are felt to be authoritarian (though they can be) but that their existence changes not only accepted procedures but accepted ways of thinking. As Weizenbaum (1976) points out the existence of AI programs changes our canons of acceptable thinking away from judgment towards calculation. Each discipline fears that changes will be wrought in its theoretical fabric. Perhaps the clearest example of the effects of introducing a new technology, historically, is the case of the changes wrought by the invention of printing. A largely oral poetical tradition and a tradition of manuscript illustration more or less totally disappeared although prestigious manuscripts were often illustrated by hand until the sixteenth century. Thus resistance to AI programs need not be brought about by hide-bound conservatism or

technological illiteracy, nor even technological inadequacy. It may represent the fear of some that the use of such programs will lead inevitably to the withering away of certain of our abilities. We become even more the creatures of eye and hand and less the offspring of imagination and thought.

I have derived these objections largely from the work of Huggett and Baker from the Research Centre for Computer Archaeology at North Staffordshire Polytechnic (Huggett 1985, Huggett and Baker 1985, Baker 1986, Reilly 1985). These objections and my rejoinders can be summarized as follows (H = Huggett et al., AS = Arthur Stutt):

H: AI people use archaeology as a test-bed for ideas which are archaeologically inadequate.

AS: I freely admit that my ideas are archaeologically inadequate. It is up to archaeologists not to use them as they stand. I am advancing my system not for its archaeological prowess but because of its knowledge about argumentation. As in any other branch of science or even of the humanities argument is necessary.

H: Expert systems appear too intelligent and therefore lead to false expectations.

AS: I agree with this and hope that the necessity for my system to attempt to convince the user of the correctness of its decisions by defending them goes some way towards a solution. Graphical presentation techniques also reduce the risk of easy trickery. (See chapter 5 below for a design principle based on this notion.)

H: There are the problems associated with standardizing, abstracting and formalizing archaeological knowledge. '...the degree of formalization necessary to construct an expert system is a form of reductionism, in that the translation of knowledge from the implicit to the explicit will inevitably involve the loss of elements in the process...' (Huggett 1985:135)

AS: (a) This is true of the formalization of any domain by whatever formal method be it mathematical or logical. The nice thing about AI formalisms is that while they are in some ways inflexible they are perhaps more psychologically plausible than say quadratic equations.

(b) Some gains will also accrue. Some would say that non-formal methodologies are hardly methodologies at all. I do not share this belief since I feel that many of the inferences we make in natural language are more subtle and creative than any possible in a formal system. However this is again a general problem we can only touch on.

(c) The arguing system seeks to capture the actual reasoning processes of archaeologists (or whatever specialists). It does this by utilizing a range of logical

and non-logical inference methods which make up the common argument style for a domain.

H: Representing the knowledge will lead to what may be called the 'fossilization' of a particular conceptual framework : 'The widespread and uncritical adoption of expert systems could result in the stifling of archaeological theory, since the encapsulation of archaeological knowledge may in fact lead to its stagnation.' (ibid:135)

AS: Huggett may be right about most expert system shells. However a large part of the current research effort is in producing systems capable of automatically changing their knowledge base or 'learning'.

H: Not all the assumptions made in reasoning are justified

AS: Huggett writes as if human experts can do this. I feel that ultimately we must all retreat to a general (culturally determined) basis for our beliefs as well as to the citing of authorities as the sources of our knowledge.

H: Only deductive or quasi-probabilistic reasoning is available

AS: Researchers are currently working on many other forms of reasoning. These will soon be available as a part of commercially available products. For example, there is work on analogy (Gentner 1983, Keane, 1988a, 1988b, Keane and Brayshaw, 1988), on plausible reasoning (Collins 1978, Baker, Burstein and Collins, 1987) and, on abduction (Josephson, Chandrasekaran, Smith and Tanner, 1987).

H: Archaeology is not a good domain. 'Archaeology does not seem to be the natural candidate for the application of expert systems that some would believe.'

AS: But Doran (1977: 433) has said: 'Archaeology has clear attractions as a problem domain for artificial intelligence research. Many of the problems of recognition and interpretation encountered in archaeology have close parallels with classic artificial intelligence problems, notably those of scene analysis.' There is obviously a lot of disagreement here. My feeling is that Doran is right if only because the domain *is* attractive even if some of the problems remain intractable, as Huggett suggests.

While one of the most successful expert systems - R1 - operates in a domain in which the the knowledge is complete and unchanging there have been other systems in less certain domains such as medical diagnosis. There is no *a priori* reason why an expert system should not be possible in archaeology. An Argument Support Program

(ASP) functions as such a system, finding plausible rather than valid arguments, comparing arguments and finding and utilizing responses to arguments. The ideas are drawn from a study of rhetoric and should not upset any archaeological reasoning patterns since they must hold for all arguing. Something of the kind must be used in any scientific or non-scientific area. Such a system is non-directive. It works as a partner in the reasoning process and must be thought of, not as an oracle delivering its prognostications, but as the fellow you share the office with who is always willing to listen and attempt to refute your arguments. In this way I feel that my system is, as Huggett requires, dictated by the domain it is arguing about rather than the other way around.

1.4 The thesis - model, program design, program

In the pages which follow I shall present a model of argument, a design for a computer program and a partial implementation of the design. In this section I will present a brief summary of each in order to provide the reader with an outline of the path to be followed before the details of the journey are encountered.

- **Model**

Arguments can be viewed at three levels: debates, interpretive arguments which are deployed in debates and the argument steps which make up the interpretive arguments. Since the term 'argument' is ambiguous I have adopted the use of the following terminology for the three levels: argument-0, argument-1 and argument-2. While this terminology is ugly, it serves to stave off any incomprehension or misunderstanding caused by lack of clarity in the use of the word 'argument'. An argument-0 represents a step in an argument-1. An argument-1 is equivalent to an interpretation. An argument-2 represents the on-going dialogue between two or more participants each of whom deploys a series of argument-1s. The three levels and the relations between these is illustrated in Figure 1.1. As the figure shows, there are graphical equivalents for each of the three levels. In representing an argument-2, I have concentrated on the relations between the argument-1s which constitute it. For this reason the graphical form of the argument-2 only shows claims and grounds. The figure illustrates how the argument-2 can be decomposed into its constituent argument-1s and their constituent argument-0s. The expanded versions of argument-1s (interpretations) and argument-0s (graphical toulmin structures) contain

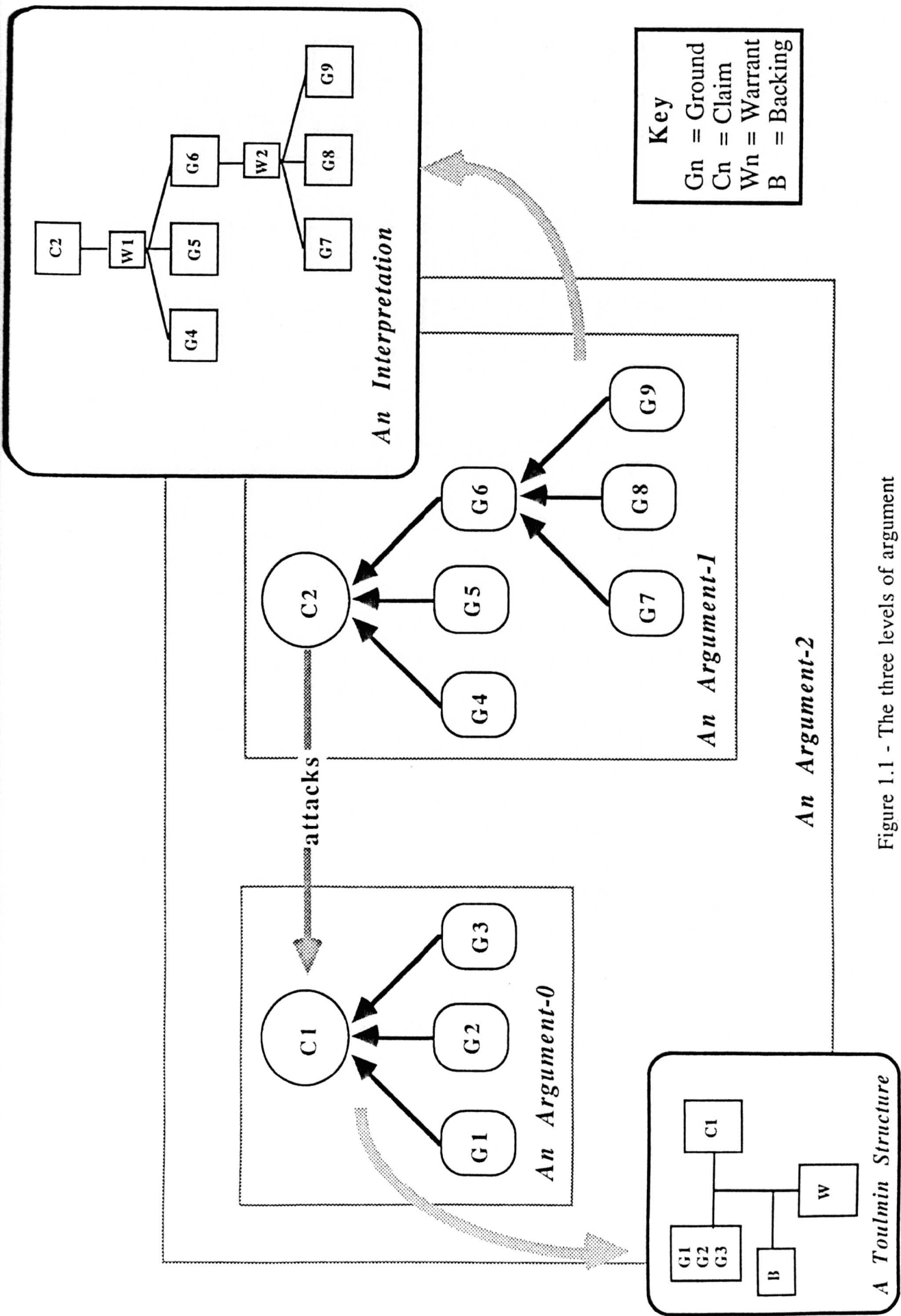


Figure 1.1 - The three levels of argument

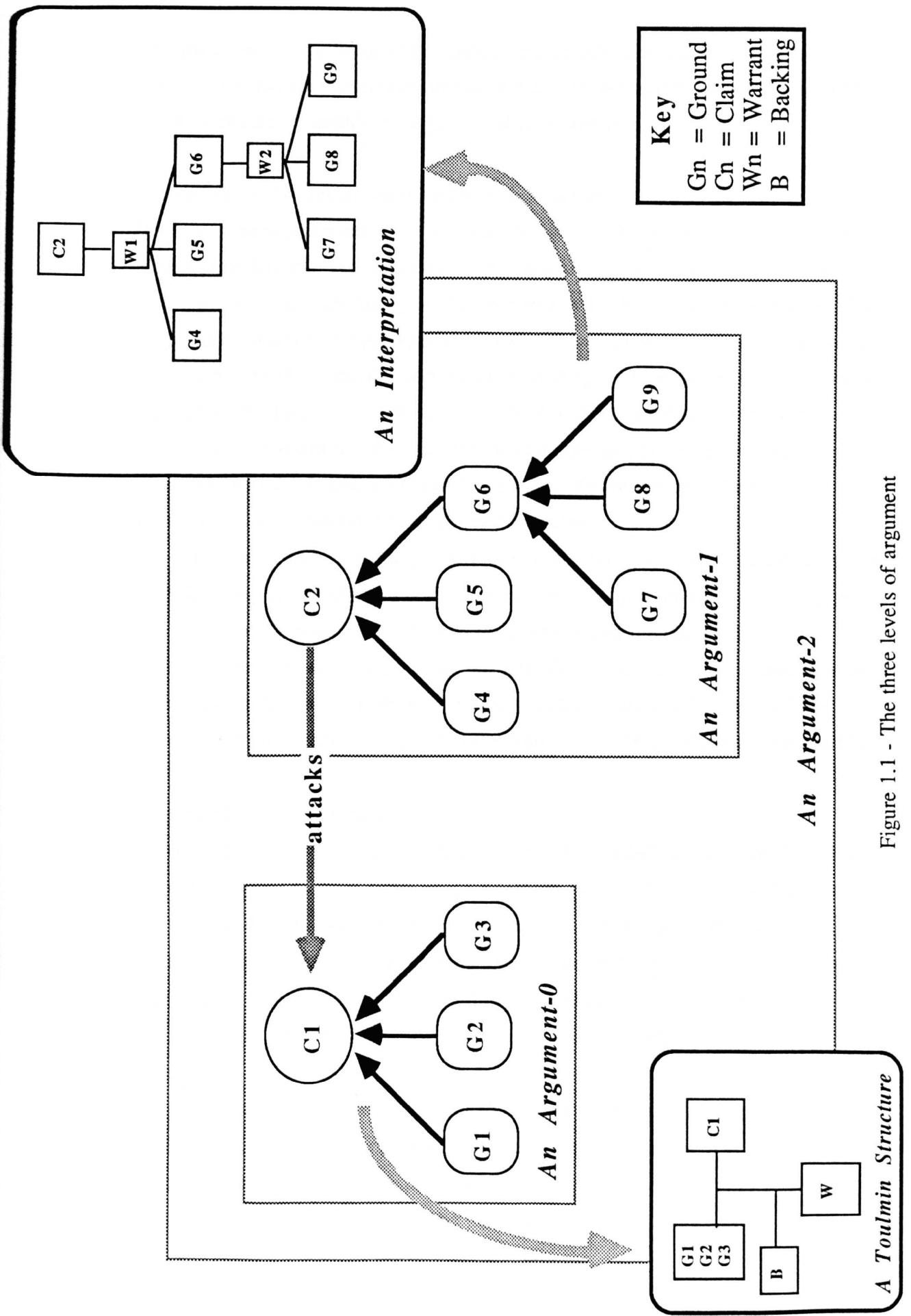


Figure 1.1 - The three levels of argument

more information. In the body of this thesis I make extensive use of graphical argument-2s and toulmin structures in illustrating various points. The program to be discussed in chapter 6 mainly uses graphical interpretations.

- argument-0s, argument steps or toulmin structures

These may be represented as in Toulmin, Rieke and Janik (1979) as a series of related propositions. An argument step is a structure which has a claim and a set of propositions representing grounds, warrants or backing. Grounds can be related to claims by warrants or derivation rules drawn from common sense, accepted as conventional in the particular domain or depending on some theoretical principles relevant to the domain. These different bodies of knowledge constitute the backing for the warrant. The warrants which compose the relations between propositions at the lowest level can be of many different types. I have concentrated on deductive and analogical linkages.

The reader will note that in the toulmin structure the warrant is a branch from the link between grounds and claim. In an interpretation, on the other hand, the warrant is a node positioned between the grounds and the claim. Toulmin structures are used here as Toulmin himself illustrates them. They should be read as showing how a warrant relates grounds to claim and how backing relates warrant to grounds and claim. The two representations are therefore equivalent.

- argument-1s or interpretations

An interpretation is taken as a tree-like structure in which a top level claim (or claims) is supported by a series of propositions or grounds. Each of these sets {grounds, claim} constitutes an argument-0. Each of the grounds may in turn provide the claim for a sub-argument. The relations between the propositions are mediated by the warrants or rules and include relations based on deduction, analogy, causation and so on. An argument-1 may be composed of a single argument-0. Whereas, in archaeology, the argument-1s are the equivalent of interpretations, in other domains, an argument-1 might constitute, for example, a mathematical proof. An argument-1 viewed as an interpretation is the result of a process of applying transformations to facts about a domain. The interpretation can be seen as the elucidation of the meaning of the data.

- argument-2s or debates

The model of argument exchange draws on two analogies: i) an argument can be viewed as a series of moves in some game, for instance, a card game; ii) an argument can be viewed as the exchange between two lawyers before a judge who then gives a verdict. The former analogy is intended to capture the following facts: (a) that an argument has moves, which are governed by rules; (b) that argument participants take turns; and (c) that the result is usually that one of the participants wins the argument. The latter analogy points mainly to the assessment of an argument which we make in judging its strength and the overall winner. Each participant has a series of turns in which claims can be attacked or supported. An argument exchange is taken to be composed of a sequence of argument-1s which may be spread over several turns in the argument. There are relations of support or attack between the claims of previous and subsequent steps. An argument exchange can be represented as a graph-like structure with argument-1s as its nodes and relations of support and attack as its edges.

A computational model of arguing or debating must also deal with the active as well as the relatively static components mentioned above. Thus the model must include components for the parsing, checking, generation, and assessment of arguments as well the response to them. A satisfactory theory of response is one of the main foci of this thesis. These active elements form part of the arguer who by making use of domain knowledge will produce an argument as described above.

While much of this model is generally applicable to many domains I shall suggest that archaeology is characterised by the use of two specific kinds of argument: argument by analogy and argument from models or theory. I have thus concentrated on these two both in terms of how they may be generated and how they might be responded to.

- Program design

The generic Argument Support Program - ASP - and the version of the program applicable to archaeology - ASP for archaeology (or ASParch) - designed in accord with this model have the capacity to store and retrieve domain knowledge as well as knowledge about how to argue. It includes procedural code for the operative components mentioned above, declarative representations of the domain, declarative

representations of strategic knowledge and assessment knowledge as well as frame like structures for argument-0s, tree-like structures for argument-1s and a network for argument-2s. An ASP (see Figure 1.2) is composed of two main modules: (a) the argument module and (b) the underlying domain knowledge base. In terms of the above model, the components of the argument module represent the operative elements (the arguer). These use the system and domain knowledge to produce argument-1s as part of an argument-2.

- The argument module

Components include control, user argument parsing, user argument checking, system argument generation and overall argument assessment. At the heart of the system will be the reasoner which can draw upon an inference engine to produce or assess arguments which can be either deductive or analogical. The system will also include a means of switching between the system and user viewpoints, a means of updating these viewpoints and of storing them. A system knowledge base will be required consisting of knowledge of viable next moves. An argument net composed of frame-like nodes for storing the argument as it proceeds will be an essential component.

In any automatic argumentation system it will be necessary for the system to have some means of responding to the user's arguments. Rather than following a simple algorithm, an ASP makes use of data-structures known as *troublemakers* to focus the system's reasoning. These troublemakers consist of a two stage process in which the actual response is preceded by a process of discovering faults in the opponent's argument thus determining the contents of the response.

- The archaeological knowledge base

The knowledge base must contain knowledge about a particular sub-domain or domains. This knowledge constitutes the factual context for any argument. Since this is so the knowledge base must be as complete as possible so that every likely topic is covered.

- Implementation

In the implementation the principal focus has been on the generation of responses to user arguments and, in particular to those which make use of analogies or depend on

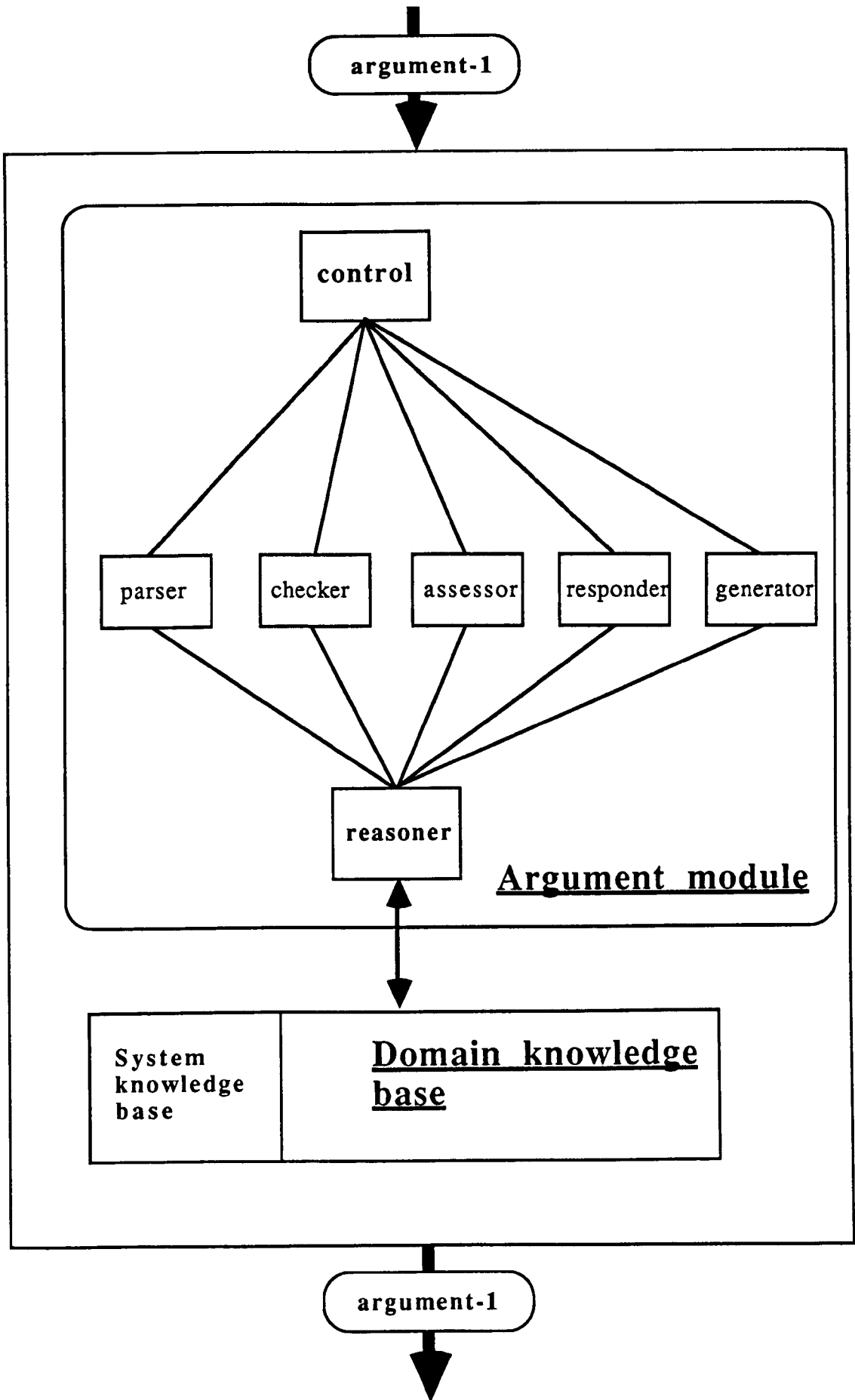


Figure 1.2 - The architecture of an Argument Support Program

various theoretical viewpoints. These can be viewed as an extension to the basic set of troublemakers. The partial implementation of the above is in terms of a constrained arguer or debater. Here the user can act as a participant in a formal debate in which each side has two moves. In the first move a statement for or against a particular topic is made. In the second move that statement is either supported or the argument of the other participant attacked. The participants take turns. This constrained implementation allows me to illustrate clearly what I envisage happening in a full arguer without worrying about all the processing tasks that that would involve. These issues centre on the notion of how, in a given field, a reasonable response to an argument is generated and how analogy and theoretical argument are dealt with. I will present a series of examples which illustrate how the system would deal with the archaeology specific tasks mentioned above. At the same time I suggest that these may be generalizable to other domains. I also suggest that debate is a sub-component of a proper free-flowing argument and that therefore, it should be possible to defend my model on the basis of the success of the partial implementation of the model.

The knowledge bases used are standard fact and rule knowledge bases. In the implemented version, the system has partial knowledge about the modern Cree Indian site described by Bonnicksen (1973), the Pueblo Indian sites described by Schiffer (1976) and Longacre (1970) and the Wessex culture of southern England in the Bronze Age described in Piggott (1938).

1.5 Examples

In order for the reader to get some idea of the possible use of an argument support program, I will outline two example arguments and suggest how a computer based system could be used. To illustrate the kind of tool I envisage I will show a) how it might tackle an argument drawn from a newspaper article and b) how it deals with a real example from archaeology.

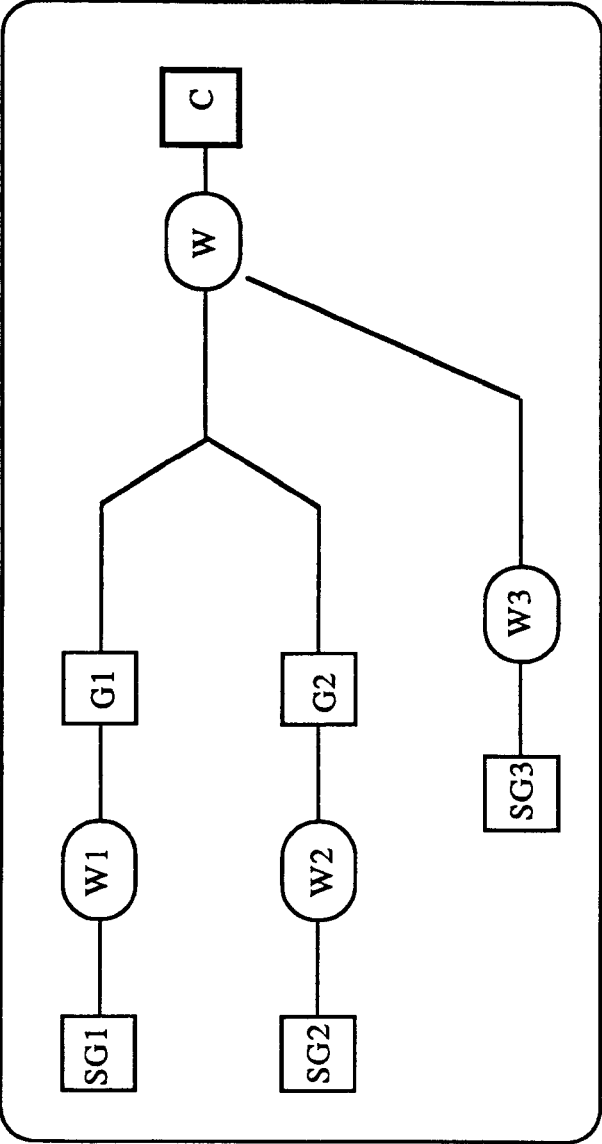
1.5.1 A hypothetical example

This example serves two purposes. Firstly, for the reader whose knowledge of archaeology is not wide, the example serves to shed light on the argumentational aspects of the argument as opposed to its archaeological content. Secondly, the example, lends some weight, I hope, to my general claim that the method I am

discussing need not be confined to the archaeological domain. This example also concentrates on the method as a technique for analysis of arguments while the second example illustrates the functionality of an arguing machine.

The Guardian (of 21st January 1989) carried an article by Paul Fussell in which he argues forcibly that the atom bombs dropped on Hiroshima and Nagasaki were necessary since they obviated the need for an invasion in which 'a staggering number of Americans', 'Thousands of British assault troops' and numerous Japanese would have lost their lives. It would be possible to analyze this article and extract all the propositions which are contained therein and work out the various relations between them. But for the illustration we can take the seemingly central claim that the atom bombs dropped on Hiroshima and Nagasaki were necessary. We can call this C. This statement rests on the ground that otherwise a great number of human beings would have been killed: let us call this G1. This claim could not on its own produce the conclusion required. Even if we are not talking about strict implication here, G1 on its own doesn't lead to C. We need at least the assumption that it is better to do something which will kill a known number of people than to do something which may kill a greater number of people. Let us call that G2. It might also be useful to extract the warrant for moving from G1 and G2 to C. Again this is not stated but we might assume that it would be of the form: If better to do X than Y and the proposed action is an X then necessary to do X. We can call this warrant W. Immediately we put the argument this way we can see that even with G1 and G2 and W we cannot derive C. We can only derive 'It is better to do X' not 'It is necessary to do X'. Other assumptions must be involved to produce C. Furthermore we can see that G1 cannot stand as self-evident. A sub-argument which has G1 as its claim and at least one ground SG1 and a warrant is needed. We might also wonder why we should accept G2. This in turn needs a sub-argument. Again we have no reason to accept the warrant W. Thus we might require an argument which has W as a conclusion and some grounds for this conclusion plus a warrant for deriving the grounds. Finally we have, for this simple argument, built up a complex structure (Figure 1.3).

We should note that this structure is *open-ended* in that any ground or warrant can require further support. It should be noticed that in this attempt, as in any attempt at an analysis of what an arguer actually says, an element of interpretation is involved. That is to say that we could have supporting arguments for the interpretation given above which we can call I1 as opposed to another interpretation



Key

Cn = Claim
Gn = Ground
Wn = Warrant
SGn = Sub-ground

Figure 1.3 - The structure of a hypothetical argument

I2. Thus, not only is the argument contained in I1 shown to be open-ended, the process of producing I1 is always provisional. The sort of analysis given above (which will be elaborated upon in the body of the thesis) is something which we may do consciously if we have been exposed to logic or informal logic. It is also something we do naturally. We need to produce an interpretation in order to respond to the argument. The production of interpretations is often thought to be subjective and that logic and related techniques give an objective means of analysis. This is only so in that they provide a means of making purely prejudicial response less possible once an interpretation is made. There is nothing in logic or in the method given above which ensures the correctness of the interpretation of the natural language statements of an argument. This interpretation will always be an uncertain, plausible and provisional process. At the same time certain elements become clearer when exposed in the above manner. This clarity can be enhanced by a computer program which automates the analysis by prompting the user for the necessary elements in building up an argument.

The representation and manipulation of the hypothetical argument discussed above would be accomplished as follows. The argument-0s would be stored as a set of frames, argument-1s by a tree of nodes each representing an argument-0 and argument-2s by a network of linked argument-1s. Thus, for example, the above could be represented at the lowest level by what I have called a toulmin structure, i.e. a frame structure (Minsky, 1975) with slots for various different kinds of proposition:

```
name:  fussell-top
claim: the atom bombs dropped on Hiroshima and Nagasaki
       were necessary
grounds:
  (1) it is better to do something which will kill
      a known number of people than to do something
      which may kill a greater number of people
  (2) a great number of human beings would have
      been killed during invasion of japan

warrant:
  If better to do X than Y and the proposed action
  is an X then necessary to do X
```

Manipulation would include the checking of elements of the network against facts

stored in a knowledge base by means of a reasoner which is capable of a full range of human reasoning abilities i.e more than deduction. The argument might be displayed as in Figure 1.4.

The system might prompt for, or search for (if input is text of whole argument), the elements which make up the above. Similar arguments could be produced by the system by interpreting its knowledge base (and ideally slanting this towards a particular ideological stance). The system would be able to make use of overall strategies and rule-like ways of responding to select an appropriate response - for example, Figure 1.5 - and knowledge about what counts as a good argument in assessing the strength of the current argument.

1.5.2 A real example

The following example is intended to give the reader a foretaste of what a program based on the model and design I shall introduce in the following pages is capable of. Similar examples will be presented in chapter 6.

The example is based on the *Millie's Camp* domain (Bonnichsen, 1973). Bonnichsen describes in this paper an archaeological experiment in which archaeologists interpret a modern Cree Indian camp and check this against the actual use of the camp as given by Millie, the occupant. The archaeologists get some things right but make several errors in interpretation. The paper as a whole is evidence of the need for some means of representing and manipulating archaeological inferences so that errors of interpretation are minimized. The particular importance of the paper for my work is that Bonnichsen gives clearly the conclusions arrived at and the justifications for these conclusions. The example I am concerned with here is based on a small piece of the interpretation where it is argued that one particular activity area (or part of a site in which significant activities were performed) is used for hide working because of the discovery of a frame construction found in the activity area.

In my reconstruction of the example the machine acts as Bonnichsen's interpreter. As we can see from Figure 1.6(a) (redrawn from a Macintosh™ output screen) the system puts forward the argument that the area is used for hide working because it has a feature (feature11) whose use is hide working. The user responds as shown in Figure 1.6(b) by suggesting that the activity of the area is cooking because there is an analogy to feature11 whose use is cooking. To this the machine responds in turn by

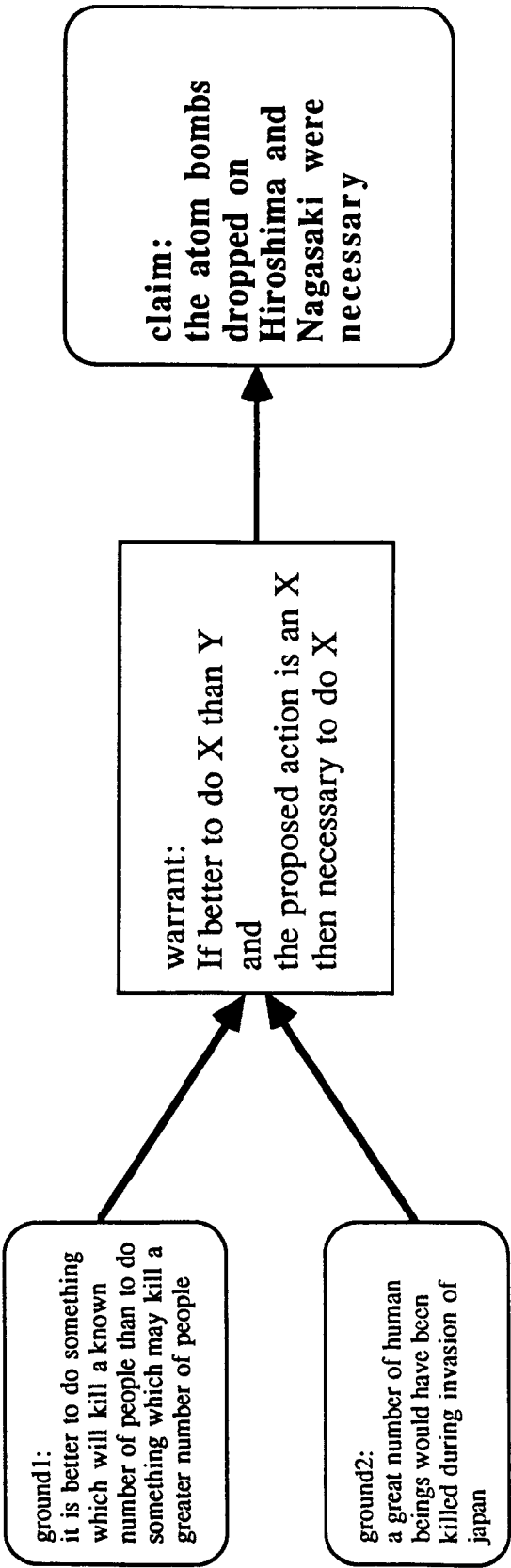


Figure 1.4 - A hypothetical argument

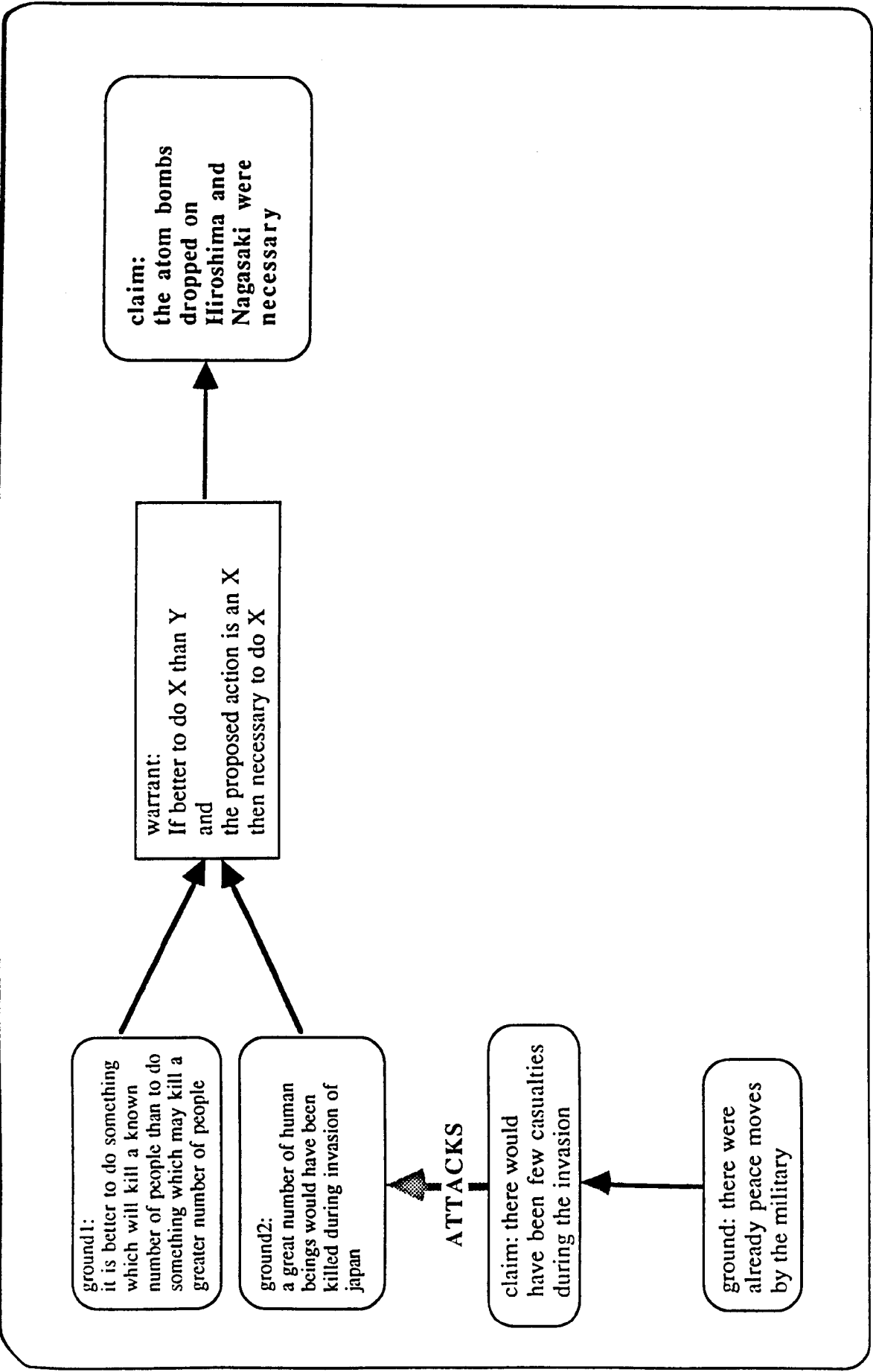


Figure 1.5 - A response to the hypothetical argument

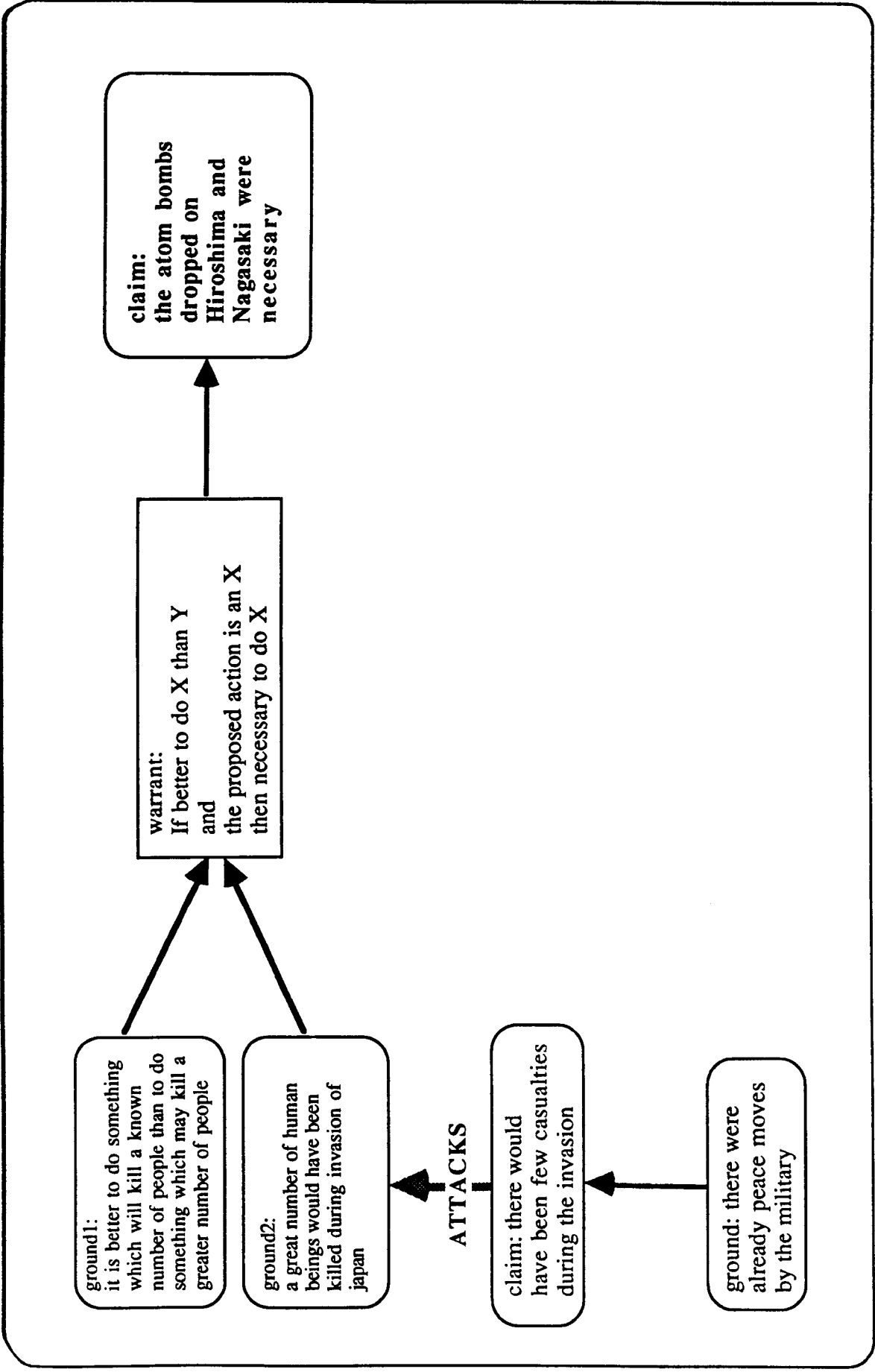


Figure 1.5 - A response to the hypothetical argument

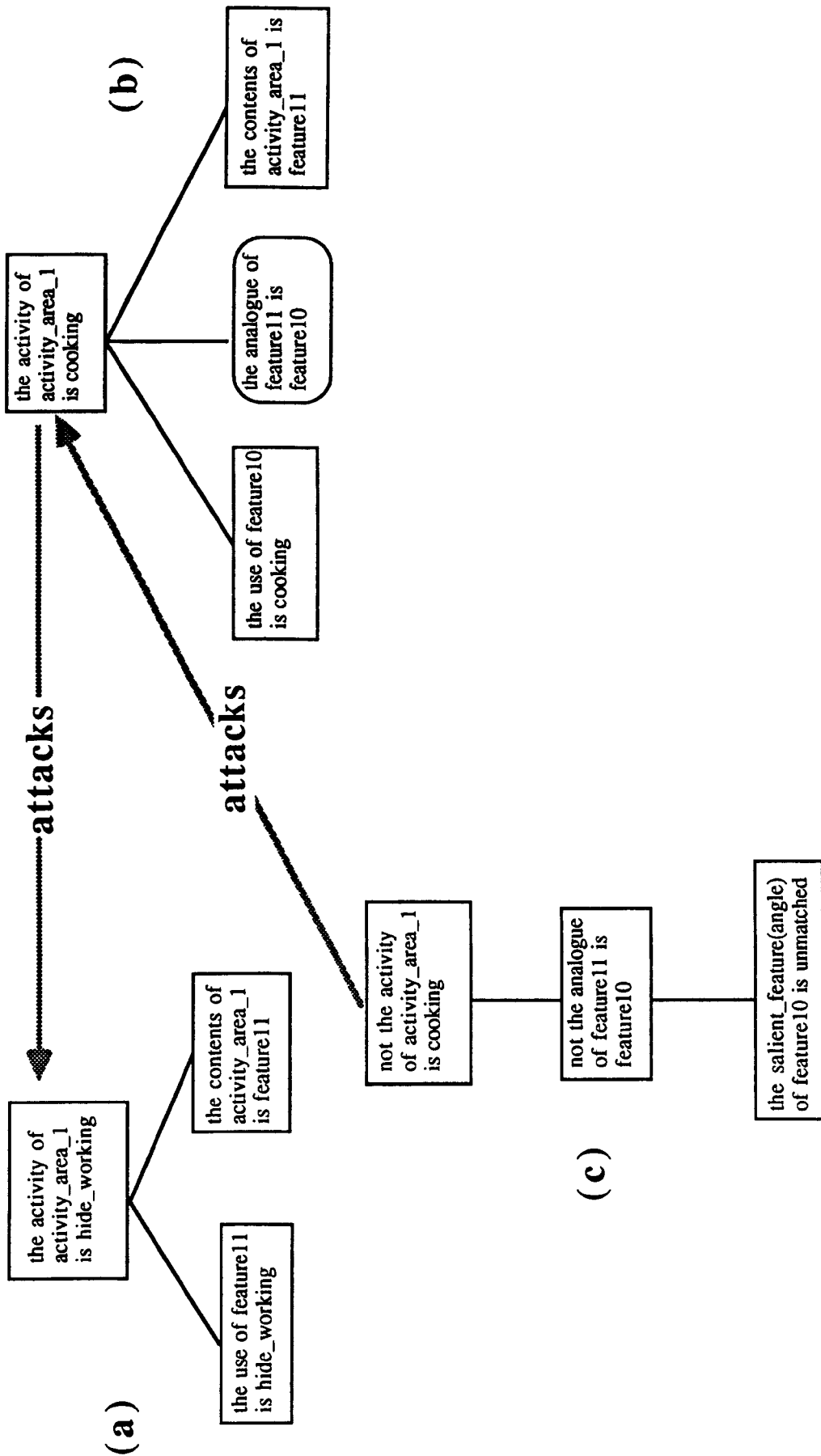


Figure 1.6 - An ASP generated argument

providing a partial critique of the user's argument as shown in Figure 1.6(c) in drawing attention to the fact that the analogy with feature10 relied upon by the user is suspect because it does not depend upon a salient attribute i.e. the angle. The user argument here is not derived from the Bonnichsen paper but created for the example. Feature10 can be thought of as a spit, that is, a construction for cooking. The hide-working frame is also a construction but one in which the angle between it and its support is of crucial importance.

Figure 1.6 as a whole illustrates part of the argument network which is produced by this exchange and includes the relations between the user and system argument-1s (although the graphical display of argument networks is presently unimplemented). Each node in this network is similar to the hypothetical node given for the Fussell argument above and is represented in the following way:

```
arg1 instance_of argument with
  claim: (the activity of activity_area_1 is cooking),
  grounds: [(the analogue of feature11 is feature10),
            (the use of feature10 is cooking),
            (the contents of activity_area_1 is feature11)]
```

The most important aspect of the machine's functioning in responding to the user's argument is a set of rules which I have called *weakstatus* rules. These determine whether any of the nodes in the user's argument are weak in some way and represent the implemented version of the first stage of the *troublemakers* mentioned in section 1.4. In the example given, the node which contains an analogy is identified as a potential weak spot. This weak spot is then tagged as an *analogy_ground* and passed to another set of rules for dealing with the weak spots. The particular rule which is fired in this instance is the rule which detects the misuse of an analogy because the analogy does not match on an attribute which is regarded as salient in this domain.

As we can see from these examples, the computer has a role as a displayer of arguments, an expositor of assumptions, an imposer of a particular analytical and interpretational methodology and as a responder to arguments. A computational arguer provides a means of representing an individual argument as well as the interrelating arguments which form a debate. The system can display the argument in graphical form, clearly distinguishing its components. The system can prompt for

elements in an argument. By this means it provides an invaluable tool for the revelation of concealed assumptions. The passive analysis and display of the argument has the effect of exposing all the interconnections between claims and other claims. The active response has the effect of highlighting weaknesses in the support for a claim and in providing some idea of the possible ways these might be attacked. The system can represent the grounding of the argument in backing theories, common sense or ideology thus allowing for the display of a particular viewpoint on the facts. The system can produce an argument insofar as it can produce an interpretation and embody strategies for responding to an argument.

1.6 Conclusion

In summary then this thesis will:

- outline a model of argumentation and its relation to concepts such as justification, explanation and so on;
- show how this theory provides a model for the design of interfaces for expert systems - especially for users in the humanities;
- show how the program designed in accord with this model can be seen as both
 - a) a tool to aid argument production and assessment - focus on mode;
 - b) as a natural means of interaction with a knowledge base - focus on content.

Insofar as argumentation is the meat and bones of many of our discourses both in our everyday life and in our specialized pursuits (academic and non-academic) this attempt is important. However, for the same reason, the attempt must ultimately fail. In order to be able to "argue" as a human can a machine would need to be the isomorph of a human being. Arguing is not something we do independently of our other capacities and attributes. Thus the model I will produce must of necessity only approximate to a human arguer. My attempt must be seen not as an attempt to deal with the whole of arguing but to attempt to model some elements of the process. In this it is no less of interest than other attempts. However the computer enforces a degree of objectivity on the attempt which is of importance and, when coupled with artificial intelligence techniques of representation and manipulation, a model and a

simulation of arguing can be produced which, while formal in that a computer language is used, is also human-like in taking account of more than the merely mathematical aspects of logic. I am aware that computer technology is in itself determined by particular social and political constraints. But then this is true of any form of expression which I might attempt to use. The computer, like natural language, can be used as a means of exposing its own determining factors. The automation of arguing processes also allows the possibility of exposing and manipulating the processes which shape our arguments.

1.7 Guide to the reader

Chapter 2 contains my characterization of the domain of archaeology in more detail than that attempted above. I point out the aspects of argument which archaeology shares with other domains and those which may be regarded as domain specific. In particular, I will discuss the use of argument by analogy and argument from theoretical principles or models.

Chapter 3 contains a review of related literature. This falls into two sections: (a) work on explanation; (b) work on computational models and implementations of systems for argumentation. I have included the review of explanation capabilities since I feel: firstly, that explanation and argument are closely related; and secondly, that, historically, argumentation represents a natural development of "explanation" capabilities.

Chapter 4 presents the three level model of argumentation in greater detail and discusses how arguments are parsed, generated, assessed and sustained. The model of argumentation includes a model of interpretation which derives from the work of Jean-Claude Gardin.

Chapter 5 presents a design for a program based on the theoretical principles delineated in chapter 4. The design concentrates on its function as a tool for more effective arguing. The design and its subsequent implementation represent a means of tightening up and testing the theoretical principles of chapter 4.

Chapter 6 gives details of the partial implementation of the model as a constrained

debater. The program ASParch is presented in terms of the knowledge structures required and the procedural code for manipulating these. Example runs from this system are also presented and analysed.

Chapter 7 discusses the application of ASParch to other domains. These are: a) literary criticism; b) legal reasoning; and, c) Darwinian theory

Chapter 8 is the conclusion. Here I discuss what has been achieved, the failures, limitations and way ahead. This chapter includes my rationale for thinking that the representation of a complete arguer is not possible.

-o0o-

Chapter 2 - Archaeological reasoning

Argument: Archaeologists have made use of artificial intelligence techniques in a number of ways: to model archaeological reasoning; to simulate social change; to act as tools. An ASP acts as a model and as a tool and therefore encompasses much of previous work in the application of artificial intelligence to archaeology. In order to model archaeological reasoning correctly we must characterise it. Archaeological reasoning is characterised as (a) making use of analogy and (b) appealing to theories.

This chapter has the following structure. I begin with an examination of current work in using AI programs in archaeology, pass to a consideration of archaeological reasoning, the use of analogy in archaeology and archaeological theory, present a brief overview of the main domains examined in this thesis and end with an example archaeological argument.

2.1 Current work in using AI programs in archaeology

It is not surprising that AI programs should be in use in archaeology. This is one of the disciplines that has made extensive use of computers in data analysis especially since the advent of the 'New' archaeologists and their desire for a more 'scientific' approach. More recently, an interest in simulations (see Sabloff, 1981) has developed into an interest in simulations based on models drawn from cognitive science and AI. This has provided one strand in the interest shown in expert systems by archaeologists. Archaeologists are interested (a) in modelling their own reasoning processes (b) using models derived from cognitive science for simulations of cultural change and (c) in the increased efficiency such systems might bring their work. It is possible that these motives might pull against each other but they surely represent the sort of motivational tangle which brings about all scientific change. I will give examples of each of these uses of AI within archaeology.

2.1.1 Models of archaeological reasoning

This use of expert system techniques is most in evidence in the work of the group of French researchers influenced by the ideas of Jean-Claude Gardin (1980, 1987a, Lagrange and Renaud, 1985, Francfort, 1988). I shall discuss the general model of reasoning employed by Gardin in Chapter 4, confining myself here to a discussion of the nature of the systems produced.

Gardin's ideas have resulted in a great number of programs which can be said to model the expertise of archaeologists in particular specialist areas. These have been discussed in Gardin et al. (1987), a compendium of research results, in which models have been developed for the interpretation of hellenistic pottery, iron-age metallurgy, cypriot pottery, roman amphorae, medieval structures and turkish stelae. Typically these authors proceed by taking a text or texts, analysing this in terms of the logicist analysis and encoding this in a form suitable for interpretation by a production system. Thus, these authors are not attempting to produce systems which can be used in day to day contexts. For this reason I have distinguished them from application systems (see section 2.1.3 below).

According to Lagrange and Renaud (1985) logicist analysis is a general technique for the analysis of reasoning patterns in scientific or non scientific discourse. In Gardin (1980) the technique is applied to what Gardin calls *constructs*. By this he means the written texts produced by archaeologists, ranging from catalogues to theoretical interpretations of finds. Basically the technique depends on analysing the piece of discourse into three major elements:

- the initial set of data which the author selects to fulfill his goal in writing. This goal can range from the purely descriptive (inventories and catalogues) to the interpretative. Lagrange and Renaud refer to this as the Po propositions, Gardin as the 'raw materials, M'. Gardin thinks the propositions derived at one level can form the material for operations at another level. For instance historical inference can depend on propositions produced by classification.
- the terminal propositions - Pn propositions - which represent the endpoint of the reasoning process whether this is simple taxonomy or interpretation.
- the intermediate 'data and operations involved in the transition from M to P'

(Gardin) the 'intermediate propositions' (Lagrange and Renaud). The latter's propositions are the textual representation of the operations discerned by the former in his analysis of archaeological constructions.

Perhaps the best known logicist analysis is that of Gardin (1980) involving a stela of the Seldjukid period in Turkey (pre-Ottoman). This analysis can be represented graphically as in Figure 2.1. In this figure a schematic representation of the stela is presented and the the different propositions which make up the interpretation are clearly shown. This example has been implemented by Lagrange and Renaud (1985).

Lagrange and Renaud give an example of the technique which is concerned with the identification of a medieval structure. The Po propositions are in this case the descriptions of the structure e.g. 'The interior dimensions of the building are about 10m X 12.70m'. (Zadora-Rio 1982, quoted in Lagrange and Renaud). The terminal proposition (Pn) that the structure is a medieval walled garden. The intermediate propositions (Pi) consist of the arguments which connect the above together. For example 'On account of [propositions] 2, a and b, of 1, a,b,c, LPG [Le Plessis-Grimoult] definitely resembles a fortress such as those built in Philippe Auguste's time'.

The basic claim is that this is an adequate technique for the capture of the discursive reasoning patterns evidenced by archaeologists in their written texts. This is undoubtedly so. However, as they admit themselves, the technique is of general applicability and therefore does not capture anything specific about archaeological reasoning. More importantly the technique does little to capture the sort of reasoning actually employed by archaeologists as they carries out the various techniques of classification and interpretation. Gardin concedes that the analyst will often have to supply the intermediate steps. Thus models derived from this sort of analysis invite the criticism that they reflect an artificial reasoning pattern derived from the analysis itself rather than from the archaeologist's reasoning patterns. However, as we saw in chapter 1, any analysis of an argument is itself a reconstruction and an interpretation. Thus the technique is a perfectly valid means of simulating archaeological interpretation.

There are interesting parallels between Gardin's analysis of the different sorts of reasoning possible in archaeology and the "model of expertise" approach to knowledge acquisition of Wielinga and Breuker (1986). Gardin's *operations* are similar to the *tasks* of Wielinga and Breuker. There are also parallels with Clancey's work on

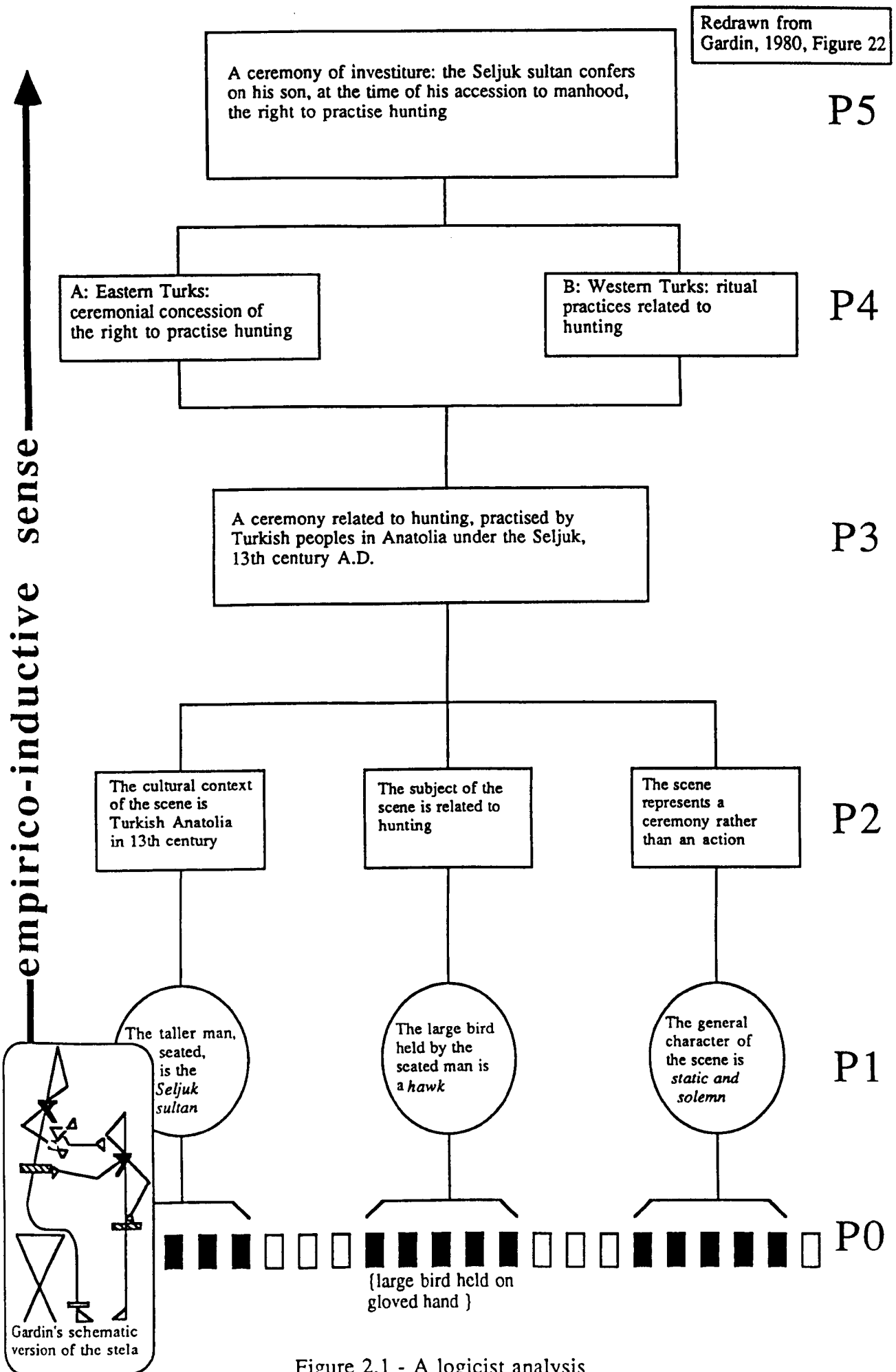


Figure 2.1 - A logicist analysis

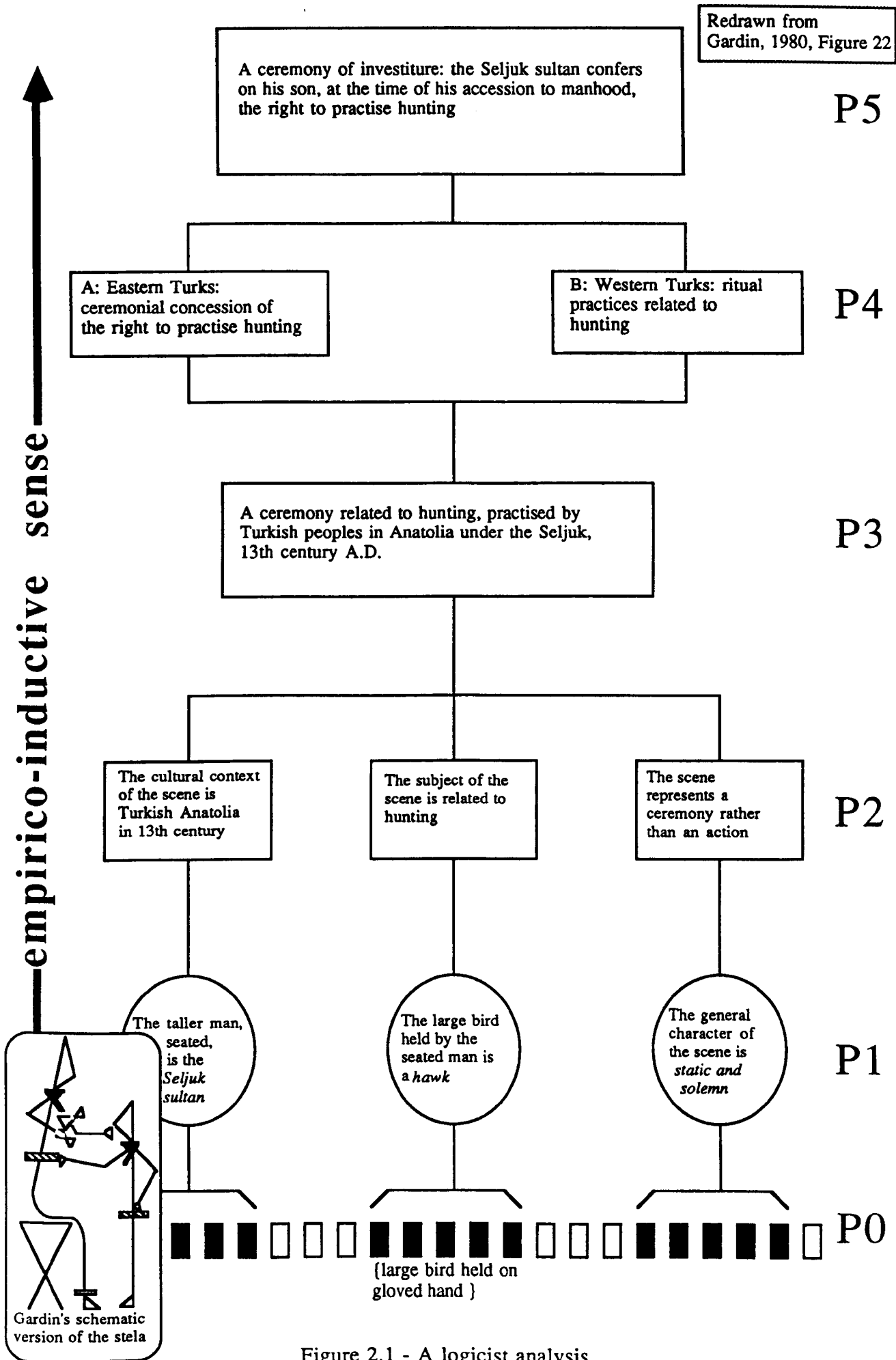


Figure 2.1 - A logicist analysis

Neomycin (1985a) and the approach of Kuipers and Kassirer (1984) in which they suggest that precise scientific models from the literature can be transformed to produce computationally viable causal models without the use of experts.

2.1.2 Models derived from cognitive science

It is here that archaeology comes closest to the interdisciplinary area known as cognitive anthropology (Holland and Quinn 1987, D'Andrade 1981) in which cultural knowledge (e.g. 'folk psychology') is seen as having a strong influence on our cognitive abilities. An example of the sort of work I have in mind here is Doran's computational model of socio-cultural systems (1982). It is obvious that if archaeology is to *explain* as well as categorize its artifacts then these must be integrated into a theoretical framework for socio-cultural change. Doran attempts to model these changes in terms of the interaction of human beings. He thus draws on studies from cognitive science as well as of socio-cultural change in order to formulate a model which makes possible an '...understanding of the mechanisms of belief or meaning in sociocultural systems..' His model is thus radically different from previous models based solely on the computing of mathematical equations. Doran's system embodies the following four hypotheses (1982: 378):

- (1) The structure of a sociocultural system reflects an inherent goal, which is the optimal exploitation of its environment.
- (2) The goal of the sociocultural system derives from the goals of its component actors. The system's structure therefore reflects the need to satisfy the actors' individual goals in spite of their spatial distribution.
- (3) The structure of the system also reflects and derives from the nature of the cognitive processing of its actors: notably mechanisms of (social) perception, goal achieving, and knowledge manipulation.
- (4) The need for cognitive economy is a major determinant of an actor's cognition. (By 'cognitive economy' Doran means the best use of cognitive abilities e.g. using previous though only partially relevant solutions to save effort.)

It is important to realize that Doran's system is not yet implemented. It is easy to see why from the far-ranging nature of the above hypotheses. To test these Doran develops a model in which a set of communicating processes (or actors) models the socio-cultural system. In the model, one actor corresponds to the environment. Communication between actors and the environment and actors and other actors is

limited in a realistic way. The environment is exploitable and can communicate its material beneficence to certain actors. Each actor's goal is to acquire, from the environment, energy resources. In order to do this it has knowledge of the environment and the other actors and can make plans. If such a system were to be set in motion, Doran predicts that its behaviour would be to show cooperation between the actors. This cooperation will lead individuals to formulate particular kinds of plans which have the effect of continuing the cooperation within which specialized activity would take place. Eventually aggregation between groups of actors will take place. Thus a 'pattern of contracts' will develop.

This seems to be a sophisticated research direction, combining as it does elements from simulation and cognitive science. If the proposal works it would represent a development not only in archaeological modelling but also in AI. The main danger is again that archaeological theory becomes constrained by alien notions from AI. Surely however this is better than the current scarcity of coherent and fruitful models? Doran's work also shows how a qualitative model for archaeology based on simulations of interacting agents could be developed to form the basis for the underpinning of shallower heuristics. This is related to much work on the use of 'deep knowledge' in expert systems (e.g. Fink, Lusth and Duran's Integrated Diagnostic Model, 1985).

In more recent papers (1986, 1988) Doran has discussed partial implementations of models similar to the above. In (Doran 1986) he discusses the simulation of sociocultural change in terms of a 'contract' model where actors can form contracts yielding 'benefits' which can be quantified. When the model is run the system of actors exhibits 'sudden global collapses'. Doran relates this to the notion that societies may collapse because they are too greatly integrated (in terms of the model, the system of contracts is too widespread) so that when a part of the structure collapses the whole follows. In the 1988 paper Doran reports on the EOS project which examines 'the properties of communities of agents, especially hierarchical organizations'. This work is not specifically aimed at archaeology but does have archaeological implications. In particular, he shows how it may be possible to model the emergence of complex social organizations such as the simple form of hierarchy mentioned. It is suggested that the model could be linked to archaeological studies in which hierarchies follow as a result of specialization and where central control is needed.

In general Doran's work is a stimulating application of recent ideas in areas such

as multi-agent systems and distributed artificial intelligence. As such, it represents a different approach to that presented in this thesis. However, it should be possible to incorporate the sorts of models which Doran proposes as one knowledge source within an overall system which at least in part makes use of shallower heuristics. Thus the two approaches are not mutually exclusive. Indeed, the combination of model and heuristics would represent the type of advanced expert system mentioned above. In chapter 6 I will propose an alternative means of dealing with models of sociocultural change.

2.1.3 Application models

There are very few such systems and most of these are prototypes of one kind or another. They do, however, represent the possibility of future developments. Most of these have been in the well-structured areas of archaeology needed in the *classification* of artifacts and structures. For instance, earlier work by Doran (1977) on the analysis of cemetery test data made use of procedurally oriented demons as knowledge sources in producing an interpretation. This work can be seen as a predecessor of the multi-actor model discussed above. Doran's work is distinct in not making use of production systems or rule and fact type knowledge bases. Most of the other work is of this form. For instance there is the work of Brough and Parfitt (1984) on the classification of animals from teeth. This was extended in Brough's work on the Fossil expert system for classifying fossil finds (1986). The work of Bishop and Thomas on Beaker pottery (1984) and Ennals and Brough (1982) on the recognition of earthworks both exemplify the expert system as a form of knowledge medium (Stefik, 1986). Ennals (1985) has many examples of the representation of archaeological and historical data for use in educational contexts.

Current work is exemplified by the system proposed by Baker for the analysis of bird bones. This system makes use of statistical and metrical studies in the identification of these remains. Further than that it incorporates 'deep' expertise about environmental archaeology and can emulate the work done after identification e.g. it can decide on strategies for further sampling and make inferences about, for example, diet (Hugget and Baker 1985). Hugget has proposed a system (similar to the subject of this thesis) which would act as the front end to a conventional database and attempt to find alternative explanations and test hypotheses. For instance, from the incomplete data from a cemetery (grave goods, position etc) it should be able to

**DIAGRAM ON THIS
PAGE EXCLUDED
UNDER INSTRUCTION
FROM THE
UNIVERSITY**

postulate which sex the occupants of the graves belonged to. Vitali and Lagrange (1988) have discussed an expert system which will assist archaeologists in the interpretation of data from archaeometric studies (i.e. the dating and measurement of archaeological artefacts using scientific means).

2.2 Archaeological reasoning and analogy

It would seem (a) that archaeological reasoning is more usually non-deductive since archaeology is at best a semi-formal discipline (as was suggested by Huggett in chapter 1) and (b) that theory in the discipline is labile since it is at a fairly early stage in its development. I do not suggest that either of these are fixed and permanent attributes of the discipline. Archaeology may well develop into a fully formal discipline such as physics. However, as it stands, archaeology is a miscellany of specialisms drawn from other disciplines such as history, geography, particle physics, surveying, natural science and geology. Attempts by the 'new archaeologists' in the States to produce a theoretical archaeology (e.g. Schiffer, 1976, Binford, 1983, 1987) have only been partially successful in combining ideas from yet another discipline - systems theory - with this interdisciplinary mish-mash. This section will (a) discuss the use of analogy in archaeology and (b) discuss the use of theory in archaeology.

2.2.1 Analogy

Analogy has played a large part in the reasoning of archaeologists. For instance Wylie (1987:6) says, '...archaeology has been perhaps uniquely dependent on analogical argument..' However it is important to note that the primary use has been in determining the use or function of something or the mechanism by which something is achieved as the following examples show. I have indicated the analogy in italics.

Analogy frame construction on this site :: frames on other Cree sites

The frame construction is similar in shape and size to frames used in contemporary Cree camps for hide stretching. (Bonnichsen, 1973)

Analogy pots :: persons

We shall argue that pots "are" persons and that concepts of the body are closely related to and partly determinative of decorative expressions on pots... (David, Sterner and Gavua, 1988)

Analogy early Neolithic boats :: Eskimo boats

It seems likely therefore that the British early Neolithic settlers used skin-boats... true sea-going skin-boats... survived into recent times in the Arctic circumpolar region... (Case, 1969)

Analogy archaeological smudge pits :: hide smoking smudge pits

The correspondence in *form* of smudge pits as known archaeologically and of hide-smoking smudge pits as described ethnographically is essentially perfect. (Binford, 1967)

Analogy stone gorgets :: pottery making implements (ribs)

The point ... is the striking similarity of the contemporary ribs and prehistoric "stone gorgets". (Curren, 1977)

Other domains also make use of analogical reasoning in the production of hypotheses. However, given the paucity of the data available to the archaeologist, analogy perhaps plays a larger part here than in most domains. Even so, because of the bad press it gets from philosophers, archaeologists have always been uneasy about this use. Thus on the one hand we have a sub-field called ethnoarchaeology (Stiles, 1977) which is concerned to make use of analogies between the culture of living peoples and those of prehistoric date. For instance, one can make analogies between modern and prehistoric patterns of residence among the Hopi based on the pottery type clustering. On the other hand Gould (Gould and Watson, 1982) argues strongly that such analogies are suspect:

Ethnographic analogies may be plausible and potentially testable, but they are often unscientific and are sometimes hard to distinguish from wishful thinking. Only when we try to explain anomalies in human behaviour as viewed in the context of uniformitarian relationships in nature can we posit the widest possible range of alternative behaviours to account for the material residues we deal with and proceed to test them in a scientifically acceptable manner. The answer to the question Is analogy necessary is thus no - or only sometimes, as long as we do not mistake it for cultural uniformitarianism or use it as a substitute for the kind of uniformitarianism S. J. Gould regards simply as good science.

There is perhaps a certain lack of awareness on the part of some archaeologists that work in the philosophy of science has advanced beyond the logical positivist models the above seems to exemplify. As Hesse and others have shown, analogy (and metaphor) form the basis of much of what can be seen as *good* science (Hesse 1980). Wylie reflects this when she suggests (ibid), that there is '...a widespread recognition among post-positivist philosophers of science that analogical inference plays a central role, not just in the formulation, but in the testing of scientific theory, the heartland of scientific rationality'.

Wylie has put forward suggestions which attempt (a) to make the use of analogy in archaeology more viable and thus (b) overcome the fears of writers like Gould.

(1) She suggests that simple analogy should be backed by considerations of *relevance*. An analogy is relevant if a piece of background knowledge exists which allows conclusions to be drawn about similarities between a base and target object. For instance, the piece of background knowledge in the case of the Hopi mentioned above may be that there is a continuity over time between historical and prehistorical Hopi culture. (I will discuss the notion of historical analogies in chapter 4.)

(2) Wylie also argues that analogy is given its viability by being accepted as ultimately grounded in *deductive reasoning*:

The conclusion I draw concerning these general arguments is that although analogical inference certainly comprises a loosely defined type of inference strategy, it does not seem plausible that its warrant is as analogical. Its warrant is that it approximates, to one degree or another, a valid inference from general knowledge of determining structures that link known and inferred properties. [ibid pg 5]

In this view analogical argument derives its validity from possible corresponding deductive arguments which make use of 'determining structures'. I take these to be much the same as Gentner's second order relations (1983) which structure or constrain the first order relations of potential analogues. An example would be a second order attribute such as *leads_to[has_wealth(person), has_expensive_burial(person)]* which expresses the notion, true of some societies,

that the having of great wealth leads to the displaying of it after death. An analogy between two cases would be more powerful if it could be shown that the first level relations or attributes (such as *has_wealth*) were generally constrained by second level attributes (such as *leads_to*). Thus it could be argued, by analogy with a contemporary culture in which the *leads_to* relation is true, that since the prehistoric culture shows evidence for expensive funeral rites therefore this culture contained individuals with large amounts of accumulated wealth which was expressed in their funerary monuments. If this is what Wylie means, then it seems unlikely that such general constraints will be found. We could easily imagine occasions either where wealth is not expressed in the above manner or where expensive funeral rites are provided for poor individuals (e.g. Gandhi).

The deductive approach to analogy has been worked out in computational terms by Davies and Russell (Davies and Russell 1987). However, if they are right in principle, such arguments may not yet be possible (since our knowledge is limited) or humans may never be able to deploy them in reasoning about the archaeological evidence. Psychologists have long argued about whether we think deductively or analogically. It may well be that humans only rarely make use of purely deductive inferences in arriving at conclusions while the path they have followed may sometimes be summarized in terms of a deductive argument. Given this and the admitted sparseness of the archaeological record, any system concerned with aiding the archaeologist in finding arguments to express his or her interpretation of the evidence must rely on non-deductive reasoning at least some of the time. The deductive form of the analogy must often remain an ideal to which inferences both of man and machine correspond only partly.

Branigan makes use of an analogical argument which may help to clarify these notions (Branigan 1985) - although there are no second order relations here, only background knowledge about architecture. He argues that since the Newgate in Londinium and the East gate at Lincoln are of a similar architectural type to the main gateways at Verulamium therefore they are of the same date.

A similar date, early in the third century, is firmly fixed for the double-portal Newgate at London and the East Gate at Lincoln which was remodelled along similar lines to the London and Chester gates at Verulamium c AD 210-230.

The gates at the other roman cities are relevant because we know that architectural features may be widely dispersed geographically while being temporally limited. Thus similarity in style of building can allow us to conclude about the date. This approximates to a purely deductive argument in which the grounds are

- a) the attributes of Newgate and Verulamium's main gates,
- b) the similarities between Verulamium and Londinium and
- c) a general rule which allows a conclusion from the known attributes and the similarities to other attributes

and the claim is the attribution of the known date of Londinium's gates to the Verulamium gates. The problem is, of course, that it is more or less impossible to find law-like generalizations such as c) in domains such as archaeology.

In an ASP, analogy will be dealt with using a knowledge base in which knowledge about the facts of the domain is structured in terms of objects which allow both deductive and non-deductive inferences to be made. This set-up is flexible enough to allow for any of the other possible forms of plausible reasoning (Collins 1978, Baker et al. 1987).

Wylie draws attention to an interesting corollary of her treatment of analogy in the following quotation:

Thus, the warrant acquired by particular analogical inferences will vary with the nature and level of understanding of these structures. Weitzenfeld adds to this the observation that while analogical arguments may have a common central structure, "patterns of reasoning [in the 'informal context of deployment of the analogue'] will vary from field to field". [ibid]

That is to say that particular domains will allow different patterns of analogical reasoning and that different individuals will exhibit different styles of reasoning depending on the "nature and level of understanding of" the determining structures or second level constraining or structuring relations. If Wylie is correct then patterns of reasoning or arguing using analogies may differ from field to field or indeed from arguer to arguer. I shall have more to say about this in chapter 4 below.

2.2.2 Archaeological theory

Leaps of faith are necessarily made since much of what archaeologists reconstruct is unobservable (Hodder 1984).

Given the prevalence of non-deductive forms of reasoning in archaeology in its attempt to make what Hodder calls 'leaps of faith' to the proper interpretation of the data available it is obvious that archaeological theory will not correspond to the sets of propositions (or equations) with relations of logical or mathematical implication applicable to 'hard' sciences such as physics. Archaeology as an interpretive or hermeneutic discipline will have theories which are more concerned with shedding light on some particular object or site than with general laws. Thus archaeology will, in common with other social sciences, adopt currently fashionable theories such as structuralism and marxism in order to provide a theoretical viewpoint on the data. The interested reader should consult Hodder (1986) for a full discussion of the different forms of archaeological theory.

There are at least two ways in which theory and interpretation interact in archaeology. Firstly there is the use by archaeologists of what may be called *extrinsic* theories. That is to say those theories which have been developed in other disciplines such as economics, philosophy and geography and which are applied to the solution of problems in archaeology. For instance, marxism and structuralism have both been used in archaeological interpretations (see Hodder, *ibid*, chapters 3 and 4). There are, secondly, corresponding *intrinsic* theories. These are the theories which have been developed within archaeology for the solution of archaeological problems. Schiffer (1976), for example, has attempted to formulate theories about the processes which artefacts undergo after deposition. There is also the so-called 'middle-range' theory of Binford which attempts to find ways of measuring the archaeological data so that the relations between the material objects deposited and the type of society that used them can be determined (Hodder: chapter 6). Indeed like many of the disciplines in humanities and the social sciences archaeology is awash with theory. We might, in fact, distinguish a third kind of theorizing in archaeology; that is, those discussions which attempt to legitimate or justify a particular brand of theory. At this level the archaeologist is acting as an epistemologist or philosopher of science. Some writers, such as Shanks and Tilley (1987: 107) regard 'theory construction as the key towards disciplinary development.' Others, such as Shennan (1988), regard the situation in a

poorer light and hope that the computer will provide a means of putting archaeology on a securer theoretical footing. As we shall see in chapter 4, an ASP provides one means of doing this.

The problem with using ASPs to model the theoretical backing for archaeological inferences is that the theoretical principles involved are rarely stated. This can be seen from the following quotes. The theoretical backing is implied rather than explicit. Indeed in many cases it is shown only by the decision to use a particular vocabulary.

ceramic material interpreted in terms of a structuralist stance

For instance, preliminary analyses of the abundant ceramic material found directly outside the entrance to Ramshög supports our contention that an expression of boundedness was an important structuring principle. (Shanks and Tilley, 1982)

grave goods interpreted in terms of marxist categories of society

...at which time burials with rich grave goods...appear in both the Elbe-Saal area and Bohemia and Moravia. From this I would infer that a hierarchically differentiated society had arisen in these areas, in which the expression of inter-individual differences at burial continued but had now taken on the role of legitimating the hierarchy through the consumption of prestige items. (Shennan, 1982)

societal change interpreted in terms of [invasions/settlement]

...settlement implies the establishment of a primitive farming system by stone-using people with primitive transport. (Case, 1969)

However, while there is the danger of misrepresenting the particular theories in other computational systems, in an ASP, where these are clearly distinguished from other domain knowledge, it should be possible to modify them easily.

ASPs are also of potential importance for the development of archaeological theory for another reason. I would suggest that any system which is to serve as an aid to theory formation in any domain further needs the ability to provide competitive arguments as a means of providing support for theoretical interpretations in the domain. Since much of the reasoning in archaeology is analogical such a system will need to provide at least competitive analogical arguments. My system can provide such arguments and can therefore be seen as a potential tool for theory building. This idea is adapted from Van Lehn et al.'s notion that competitive arguments are needed to provide support for theories in cognitive science (VanLehn, Brown and Greeno, 1984, VanLehn, 1985):

...competitive argument seems to be a tool for analyzing and clarifying the theoretical issues implicit in a computational model of a cognitive faculty.

I see no reason why it should not be applied to archaeology or any other area in the social sciences or humanities where the interpretations are concerned with what goes on inside people's heads. (Hodder (1984) thinks of archaeology as the attempt to investigate what once went on in people's heads. 'In particular, how can a scientific archaeology ... cope with verifying statements about ideas in prehistoric people's heads?') This attempt is also important to the general attempt to model human reasoning. This is because it is arguable that much of our thinking is theory derived in this way. We all carry around half-baked notions from physics, chemistry, political theory and so on. These serve as constituents in what passes for our background knowledge. The modelling of how theory interacts with domain facts in archaeology may prove of interest to AI generally

2.3 The archaeological domains

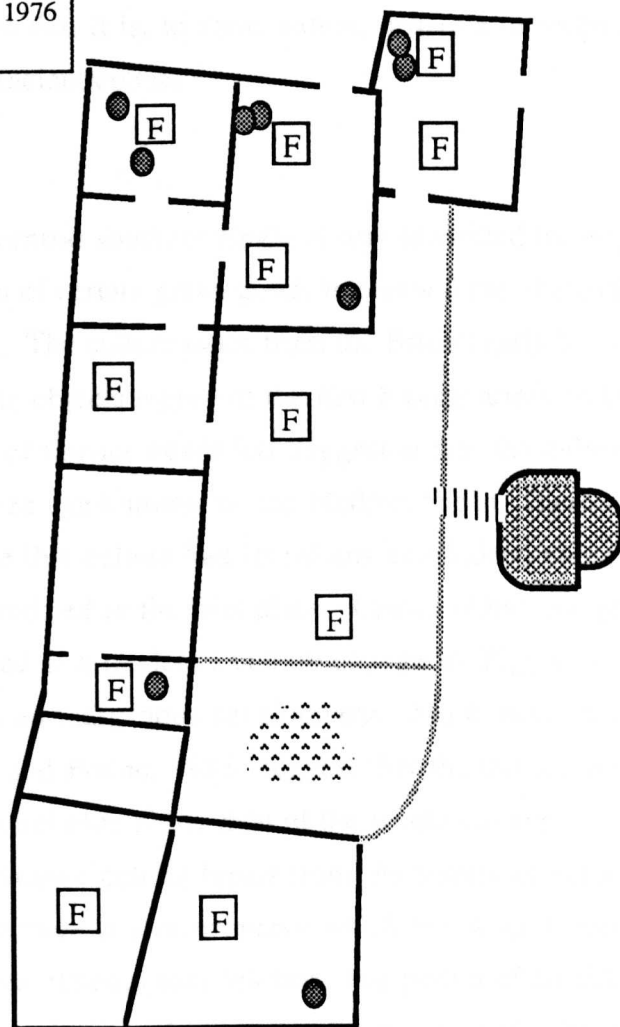
Apart from the Millie's Camp domain which has been discussed in the first chapter, and some examples drawn from neolithic archaeology and the roman period, the archaeological domains used either for illustration or implemented examples are: (a) Pueblo Indians and (b) Ancient Wessex.

• Pueblo Indians

Perhaps the most accessible account is that of Lekson, Windes, Stein and Judge (1988). Other accounts can be found in Longacre (1970) and Schiffer (1976). The Pueblo Indians belong geographically to the south-western United States and temporally to our Middle Ages. The principal cultural grouping is that of the Anasazi who in the period 1100-1300 AD built many of the large pueblos or large multi-storey many roomed buildings constructed of stone such as Pueblo Bonito in New Mexico. Figure 2.2 illustrates an idealized Pueblo site¹. These buildings and the artefacts they contain have been extensively studied by American archaeologists and are of interest for two main reasons: firstly, the state of preservation is good because of the dry conditions, hence a great deal can be discovered about the material culture; secondly, contemporary pueblo dwellers such as the Hopi seem to be historically

¹A 'kiva' is a subterranean room probably used for ritual purposes.

Based on Schiffer, 1976
Figure 7.2



Wall



Low wall



Pit



Fire pit



KIVA



Accumulation



Tunnel

Figure 2.2 - An idealized Pueblo site

related to the Anasazi so that it is, to some extent, possible to reconstruct the ritual and spiritual life of the ancient culture.

- Ancient Wessex

The Wessex culture of central southern England was identified by Piggott (1938) on the basis of the inclusion of certain grave goods in a new form of burial monument, the round barrow or mound. The culture dates from the British early bronze age around 1500BC and many of the objects represent the first bronze artefacts known in this country. The presence of faience beads has suggested that the culture was, at least, in contact with the bronze age cultures of the Mediterranean. There has always been controversy about where this culture had its origins and indeed about whether the culture was wrongly identified in the first place. Clarke (1966) suggests that the 'invasion hypothesis' used as a model of culture change by Piggott was wrongheaded and that the culture was an indigenous development. More recent accounts of the period (Clarke, Cowrie and Foxon, 1985) suggest that the culture represents the burial practices of a cultural elite rather than of the whole culture.

My interest in the Wessex culture (apart from the beauty of some of the artefacts) stems from the fact that there is a controversy which has waged over several decades about how to interpret the material remains from this period of British prehistory. The argument is also important because it concerns the question of culture change which is a central notion in archaeology since, if it is not just to be a cataloguer of artefacts and sites, it must evolve some theories about how cultures emerge and grow and finally collapse. The Wessex debate points to the fact that the arguments hinge on the different backing theories held by the participants. The interpretations produced depend on whether or not a specific model of cultural change is held. I shall attempt in chapter 6 to model some aspects of this debate. The debate over the Wessex culture also makes it clear that it is not only the conclusions which will differ from interpreter to interpreter. Given the marxist model, the purported argument about the origins of the culture might better be represented not as about the origins of culture C but rather about whether C exists at all. The theoretical backing determines what counts as facts and the concepts used to interpret these facts.

The interested reader might like to turn to Appendix II for a fuller account of the debate over the origins of the Wessex culture.

2.4 An archaeological argument

This section illustrates the notion of argument in archaeology with an argument drawn from a paper on the interpretation of Durrington Walls a Neolithic (circa 2000 BC) henge or ditched enclosure as discussed by Wainwright (1975). The argument centres around the disputed use of the henge. Figure 2.3 shows the main elements of Wainwright's argument.

I have indicated the significant claims as Cn in square brackets and the grounds for these as Gn. The argument can be displayed in diagrammatic form as in Figure 2.4. Here G5 contradicts G3 and G4 renders G1 less important. G2 is used in both arguments since the ceremonial use is accepted. This argument exemplifies many of the characteristic elements of archaeological arguments. In chapter 4 we will suggest mechanisms for rendering them computational. The elements are:

- the use of analogy
- the strategy involved
 1. attack premises
 2. replace conclusion with an alternative
- the conclusion is regarded as an explanation
- the conclusion is not certain
- the reasoning appeals to common sense

Overall, the argument represents a prime example of a well known method in philosophy - the dialectic - which derives from Hegel and ultimately Plato. Here there is a thesis - C1 - an antithesis - C2 and C3 - and a synthesis - C4.

By contrast, in the argument simulated in chapter 6 (example 3) which is about the rise of chiefdoms in Bronze Age Wessex, each side depends on a theory or model (or as Clarke 1966 has it 'hypothesis') about how societies change.

2.5 Conclusion

In this chapter we have looked at current use of artificial intelligence techniques in archaeology. We have learned that archaeology makes use of analogies and appeals to theory. In chapter 4 we shall see how a model of argumentation can be modified to take account of this domain specific characteristic. In chapter 6 we shall see how a prototype system based on this model copes with analogies and responds to the use of theory to ground an argument.

-oOo-

- The reasons why henge monuments in general are thought to have ceremonial use[C1] are, firstly, the presence of an external bank,[G1] secondly, the presence within certain externally banked enclosures of stone structures which have a presumed ritual significance[G2] and, thirdly, the absence of occupation debris[G3].
- Partially as a result of some excavations in 1972 at an Iron Age farm near Gussage All Saints in Dorset, a three acre settlement surrounded by a boundary ditch and external bank, I would not now regard the late Neolithic earthworks as anything other than secular.[C2] In certain situations, for example in areas where stock is to be confined, an internal ditch is a positive advantage [G4] and such enclosures occur for example amongst the pastoral settlements of the Iron Age in Wessex.
- A characteristic peculiar to these buildings and their surrounding earthworks is the great quantity of human refuse [G5] - sherds, stone and bone artefacts and animal bones.
- It may be suggested therefore that a consideration of those points concerning the character of the earthworks and the quantity of refuse could result in secular interpretations being placed on the enclosures and structures under discussion. [C3] There is, however, a further aspect to be considered which in the past has been termed lithicisation.
- There is therefore some indication of a period in Wessex between 1700 and 1600 bc when certain of the timber buildings at Mount Pleasant, The Sanctuary and possibly Stonehenge, were replaced by stone settings of a ritualistic nature... Its relevance in the present context is that the process should indicate some special function for the multi-ring buildings under discussion, which made it necessary for their ruins to be permanently indicated by stone settings. For this reason alone a purely domestic use for the buildings seems unlikely and a pre-eminent function within a secular context seems the explanation best suited to the arguments which have been outlined. [C4].

Figure 2.3 - An archaeological argument about Durrington Walls

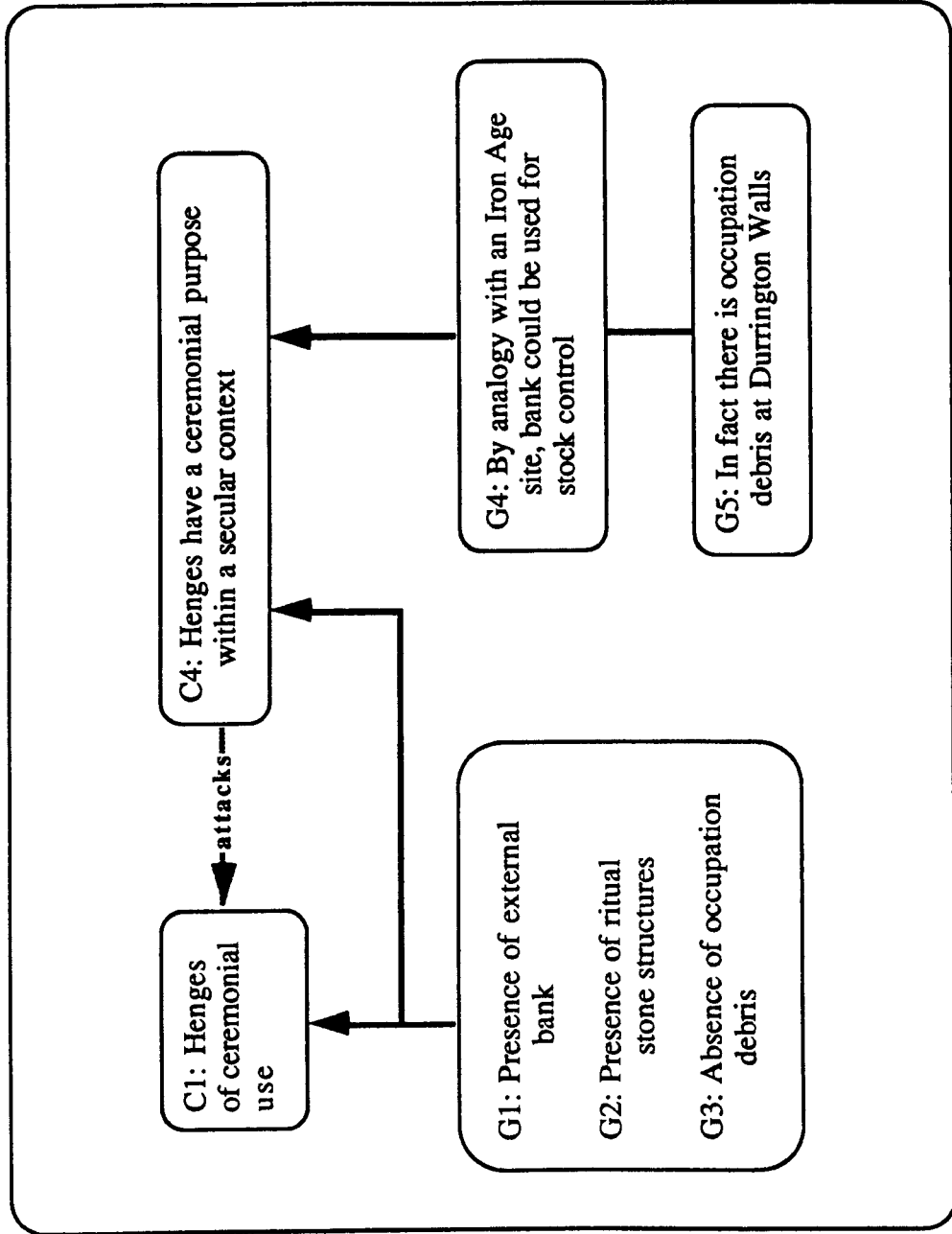


Figure 2.4 - An analysis of the archaeological argument

Chapter 3 - A review of related AI work

Argument: The historical roots of ASPs are in research on explanation capabilities in expert systems and computational models of arguing. The primary form of explanation in expert systems is justification. Justification can be seen as the provision of reasons for a claim. Thus argument intersects with explanation and justification and an ASP can take the place of an explanation capability. Explanation capabilities in expert systems have followed a line of development which results naturally in ASPs. ASPs are a means of integrating the diverse concerns of argumentation theorists with those of explanation researchers.

3.1 Introduction

This chapter aims to situate the work I am doing within the field of AI generally. There are two main bodies of research which I have been concerned with: (a) work on the providing of explanation capabilities for expert systems and (b) work on computational theories of argumentation. My own work on ASParch can be viewed historically as an attempt to apply the insights from the latter as a means of producing a better type of explanation facility. However, as the work has progressed I have become aware that the sort of system I envisage is no longer simply an expert system with a particular kind of interface but a radically different kind of system. An ASP isn't an expert system because it doesn't aim to produce *the* single correct answer to some problem. On the other hand an ASP isn't a real arguer since it doesn't have natural language capabilities and is constrained in its knowledge. Nonetheless the roots of this system are in these two areas and for this reason it is worth giving an account of their salient points. In my view of course they tend to converge therefore I have slanted the story in this direction. The convergence is however one-sided as well as perhaps fictional. While little attempt has been made by the argument theorists to examine the concerns of the expert systems researcher, s/he on the other hand has been involved in the attempt to extend expert systems so that they become interactive systems capable of graceful degradation and natural language exchanges. ASParch can perhaps be seen as a slight tangent to this work. For me the importance

of the system is not in its ability to interact in a human-like way but in its ability to exhibit system reasoning in a comprehensible manner and in a way that allows the user to interact with the reasoning. Because of my interest in arguments the emphasis in what follows will be on explanations as justifications rather than causal explanations. This restriction is not as limiting as it seems since, while causal models are widely championed as a means of providing explanations, most expert systems explanations are simply re-hashes of the system's reasoning. It is unlikely that (apart from educational contexts and specific diagnostic tasks) the user requires the sort of information included in causal models. The introduction of such models is mainly an attempt to enhance the reasoning abilities of expert system rather than to extend their explanations. This is made clear by de Kleer (1986) when he says:

Explanation is often oversold. The reason explanation is such an issue for expert systems is that the implementers of expert systems have to convince the other researchers in this field, and their own managers, that these programs work when in actual fact they don't. ... If the conclusion of the expert system was always right, or mostly right, you wouldn't need an explanation, you'd just follow it. (1986:81)

To show how similar an ASP and an explanation capability are, consider the following four functions which Wallis and Shortliffe claim are fulfilled by an expert system explanation capability (Wallis and Shortliffe, 1982). 1. A method for examining reasoning in order to debug the system while it is being built; 2. A way of assuring users of the logical nature of the system's reasoning in order to ensure acceptance; 3. A way of persuading users to accept unexpected advice; 4. Education. While the aim in examining the reasoning may be different (reply rather than peace of mind) the user of an argumentation system will also require methods for the explicit display of the reasoning of the system. The principal aim of an argument is to persuade. In fact, an argument is more directed at persuasion than an explanation is. ASPs are not directed at education in the sense of guiding students through some body of knowledge. On the other hand it is one of their functions to tutor users in an informal fashion in the use of a particular method of argument analysis and in the best responses to make. Insofar as an ASP does these things it fulfills the function of an explanation capability. However, I would argue that it goes beyond explanation just as argument subsumes explanation and justification but not the other way round. Thus an ASP is not simply an expert system though it makes use of some elements of that technology. It is more correctly thought of, more generally, as a knowledge based system in the same family as expert systems.

Before moving to a discussion of explanation in expert systems I would like to discuss the notion of explanation and argument and how they are related.

3.1.1 Explanation

Requests for explanation in everyday discourse are signalled either by a range of WH-questions (WHY, WHY NOT, WHAT IF ETC.) or by the use of the verb 'explain' or its various synonyms ('Explain yourself', 'The TUC was asked to explain its position on a pay settlement', 'Can you tell us the circumstances surrounding the accident'). Such requests are made in the expectation that the hearer will provide a satisfactory response from a range of possible responses. This range includes:

justifications - 'Why pick that one?'

puzzle-resolutions - 'Why aren't the shops open?'

fillers of gaps in knowledge - 'Why do birds migrate?'

tests - 'Explain the principles behind the use of the internal combustion engine.'

There is no clear correlation between the sorts of verbal cues we give when seeking explanations and the different sorts of explanation we get. Different sorts of requests can be signalled by similar types of questions which means that you cannot tell which sort of request is being made simply from the words used. Nor are there any clear distinctions to be made between the types of request given above. A puzzle, for example, is often resolved by filling a gap in the questioner's knowledge. A sophisticated explainer will know that a gap-filling explanation is sometimes required by puzzle-resolution question.

Given our relative lack of ability in this area coupled with the need to make subtle distinctions between the types of explanation required it is hardly surprising that we require a process of *negotiation* often before the required explanation is proffered and usually before this is accepted (cf. Kidd, 1985, Draper, 1987). Explaining is thus not a once and for all process but an open-ended interchange between human beings which ends when a satisfactory explanation is arrived at. A satisfactory explanation is one which is appropriate (to the person and the context), relevant and complete.

In expert systems, while work is proceeding on other forms of explanation, the most common form is justification - the giving of reasons for some claim which has

been made. Justifications, in general, can best be seen as a particular form of explanation which require the explainer to reveal the bedrock of reasons for a belief or action. Milton's attempt in *Paradise Lost* to justify the ways of God to man shows us another aspect of our use of the term. Here Milton was attempting to prove that God's ways were correct. But in proving these ways correct certain inconsistencies (between free will and predestination) are thereby explained. The notions of explanation and justification are inexorably intertwined. A justification of a piece of behaviour is also frequently an explanation of it. Furthermore, it is hard to entirely distinguish between a justificatory explanation and an argument.

3.1.2 Argument

There are at least two main senses of the term 'argument'. We speak of an argument between two or more people. This can range from the objective proposal and assessment of arguments as in a court of law or at a scientific meeting to the heated exchange of opinions and prejudices during a conversation in the pub. The former is often referred to as a debate; the latter as a row or quarrel. In the former any of the participants may be required to defend a particular position with reasons or supporting claims. We must all know the vertiginous sensation brought about by a hasty statement of an assumed position which we realize we cannot hope to defend. In quarrels, since what is at issue are the personal opinions of the participants, the justifications are, more often than not, self-justifications. We also speak of an argument as the proposition defended in debate by means of grounds. In the introductory chapter these were referred to as argument-2 and argument-1. In chapter 4 I will provide a model of argumentation which clearly distinguishes these two senses.

I turn now to a brief consideration of how argument has been dealt with in AI and cognitive science. This whole approach to argument and explanation depends on the assumption that arguing and explaining are part of our everyday linguistic behaviour and that, as such, they are not entirely random affairs. That is to say that there are rules which govern them. If this were not so it is hard to see how we could meaningfully conduct a conversations. Of course these rules are not the law-like generalizations of natural science. Nor are they formalizable in standard formal logic. This was a realization made by Grice (1975) in his conversational maxims (which are the philosophical backing for AI work on discourse) and before him Wittgenstein

(1953). For Wittgenstein in his later philosophy the meaning of what we say depends not as he formerly (and formally) thought on a picture-like correspondence with reality but on the situatedness of the discourse in a *form of life*. Such forms of life carry no absolute truth values and thus we have a relativistic way of looking at meaning. However, for others to understand us there must be rules. Grice showed that such rules could be made explicit in his seminal paper. Current AI work on dialogue and discourse consists mainly in finding computational means for representing these maxims (and the notion of performative utterances and speech-acts which is another child of Wittgenstein's rejection of a formalist approach). If any criticism can be made of current AI work in general it is that researchers fail to realize the resistance to formalization of many of the concepts they are dealing with. Rendering them in the Procrustean bed of a computer program might also render them essentially incoherent.

3.1.3 Explanation, justification and argument

The argument about the proper relations between justification, explanation and argument is probably, like all good arguments, totally open-ended. For our purposes we will take justifications to be the kinds of explanation most prevalent in expert systems and to be characterised as sets of propositions in which a claim is justified by a set of reasons. Taken in this way we can see that justifications are similar to what I have called argument-1s or interpretations in the introduction. Thus the set of propositions which make up an explanation could as readily form a component in an argument. Indeed any set of propositions in which one proposition is supported by the others can act in this way. Thus, other forms of explanation can act as argument-1s. Those which are composed of a set of reasons for a conclusion or which, although some of these elements are tacit, can be construed in this way. The best way to think of the relationships here is that there is a large intersection between justification and explanation and an equally large intersection between argument and explanation (largely via justification). The principal distinguishing factor between these is the purpose to which the sets of propositions are put. This purpose will guide the choice of the particular propositions which make up the set. Thus a justification will tend to produce propositions which in some sense exonerate the justifier. Explanations on the other hand are more directed at gaining or communicating knowledge. In our work this latter and the desire to get the matter straight are the only motives.

As I have said, most requests for explanation in expert systems, (when they are

not simply requests for the system to be what it is not - a human being) are requests for justifications. Why ask that question? Why make that decision? Attempts to extend the range of questions which such systems can answer may result in systems which can give proper explanations - fill gaps and resolve puzzles (the basis for scientific as well as everyday explanation requests). I see no reason why these should not be presented in the form of arguments and subsumed within a system which is capable of an extended argumentative exchange. Thus in some cases, at least, ASPs can subsume explanation capabilities. Moreover, since, as we shall see there is a move towards increased interaction with the user as a means of facilitating explanation, an ASP seems to be an ideal solution to the problem of producing viable explanations.

There is however another sense in which an explanation can be an argument. Some philosophers have argued that an explanation in science simply is an argument in a formal sense. Hempel for instance (1968) argues that there are two main forms of explanation: deductive-nomological (D-N) and probabilistic. These have come to be known as the covering law model of explanation. The former has the form

$$\begin{array}{c} C_1, C_2, \dots, C_k \\ L_1, L_2, \dots, L_r \\ \hline E \end{array}$$

Where C_n are statements describing facts and L_n are general laws and E is whatever is being explained. A probabilistic explanation takes the form of an argument from the existence of certain factors and the high probability of something given these factors to the claim that that something is very likely, where $p(O,F)$ means the probability of O given F .

$$\begin{array}{c} F_i \\ p(O,F) \text{ is very high} \\ \hline \text{makes very likely } O_i . \end{array}$$

The model can be criticised both for its narrowness and for its width. We can see from

the example of Darwin's theories of natural selection that this formulation is far too strict. According to Darwin's theories, a feature of a particular species can be explained by means of the theory of natural selection - that species become adapted over time to the environmental conditions which are prevailing. This theory undoubtedly explains the feature. However it can hardly be called a general law in the sense that say Ohm's law is in physics. Since Darwin does not argue either probabilistically or from general laws according to Hempel he has not provided an explanation. The formulation is also far too loose. There are many arguments of the above form which do not count as explanations. For example, take the following (from Kim, 1967):

All copper expands on heating - Law

This piece of copper did not - conditions

This piece of copper was not heated - E

The above is a deductive argument but the proposition E is hardly explained by it. There is also a famous counter-example in the literature originally put forward by Bromberger (discussed in Harman, 1986). According to this we can see that there is a relation between the height of a flagpole, the angle of the sun and the length of the shadow cast by the flagpole. Given any two of these we can deduce the third. However while we may accept the deduction from the angle of the sun and the height of the pole to the length of the shadow as an explanation we will not accept, say, the derivation of the length of the pole from the other two as an explanation. As Harman concludes (ibid: 74) 'Perhaps the argument provides a kind of non-causal explanation of the angle of the sun, even though it does not offer a causal explanation of why the sun should be at that angle'. The above model has also been criticised since it does not seem to apply to social or historical sciences where there are no general laws in the physical sense.

Perhaps the most important feature of the criticism of the the D-N model is that it does not include any constraint that there must be relations of relevance between the premises and the conclusion. This is a failing with traditional formal deductive systems - there are arguments which are counterintuitive mainly because of the paradoxes created by the rules for material implication (e.g. anything follows from a

contradiction). There have been attempts to supplement ordinary logic with relevance constraints (Anderson and Belnap, 1975).

Mary Hesse (1980) has proposed a modification of the D-N model in which '...theoretical explanation [is seen as] metaphoric redescription of the domain of the explanandum'. What this means is that a scientific theory can be viewed as describing events and states in a primary domain (the explanandum) in terms usually applicable to a secondary domain. Hesse's thesis is that this constitutes an explanation. Her example is the explanation of how sounds travel - 'sound (primary system) is propagated by wave motion (taken from a secondary system)'. Hesse's thesis, if correct, carries with it the appealing notion that as part of the process of explaining something scientifically we seek analogies to other domains which are already understood. This is appealing because it is something I feel we do in everyday explanation as well. And as we have seen above, at least in archaeology, analogy is an important feature of arguments as well. Hesse can be construed as showing not that explanations are not arguments but that explanations are analogical arguments.

More recently there has been renewed interest in argument as the basis for *acts of explanation* (see Kitcher, 1981).

Another approach to the question of the relationship of explanation to argument can be derived from the work on informal logics. Toulmin, et al. (1979), for instance, list several types of explanation all of which are responses to the recognition of anomalies (puzzle-resolution) and all of which can be cast in the form of arguments. For example the explanation of Jim's throat-clearing and loud talking about something else is as follows:

Claim: Jim changed the subject to avoid talking about Mary

Ground: Jim and Mary have just broken up and he is still embarrassed and grieving

Warrant: People avoid discussing topics in public which will cause pain or embarrassment.

Philosophical discussion thus reflects our commonsense intuition that explaining, arguing and justifying are closely related concepts. Nonetheless it must be stressed that there may be senses of explanation which have nothing to do with argument (causal explanation - attributing feelings to others) and senses of argument which are not explanations (not all justifications are explanations but also not all arguments are

justifications - argument may be used as a means of discovery, for instance). However the above discussion has shown that there is a wide overlap between the two in the following ways:.

- explanation and argument both include structures which are made up of statements plus some supporting evidence (justifications)
- both have two senses: a structure and a discourse mode
- both have formal and informal modes (contrast scientific and everyday explanation, arguments in a pub and philosophical arguments)

3.2 Approaches to explanation

The following section attempts to characterise four important approaches to explanation and to identify their strengths. The approaches are:

the *traditional* approach

the *sociolinguistic* approach

the *critiquing* approach

the *cooperative* approach.

It is my contention that the development of these shows a clear progression towards a style of explanation in which there is a symmetry of possible operations available to the system and user and in which data from human discourse plays an important role. The earliest systems made use of simple *Why* and *How* type explanations. This has been supplanted by systems which take account of empirical studies of human discourse and latterly which seek to emulate the two way exchanges common in everyday conversational exchanges and within which explanation is usually situated.

3.2.1 The traditional approach

This is exemplified by such programs as MYCIN (Shortliffe, 1976, Buchanan and Shortliffe, 1984) and is characterized by the provision of translations of knowledge base rules into English in response to *Why* and *How* requests. The earliest versions of MYCIN contained a relatively primitive explanation capability. Basically if the user typed 'RULE' when the system was asking a question a translation of the current rule was displayed. Later versions of the program included both *Why* and *How* commands. *Why* worked by means of a 'history tree'. This allowed the user to examine the reasoning process by progressing towards the topmost goal using a

succession of *Why* questions. The *How* command allowed the user to trace the reasoning path (that is, the sequence of rules used to produce the interim or final conclusions).

While the work done by the MYCIN team remains the foundation upon which subsequent attempts to automate explanations must build there are drawbacks in the approach. As we shall see, later researchers in the MYCIN team have attempted to overcome some of these deficiencies with greater or lesser degrees of success. MYCIN can be seen to be deficient in the following ways. Firstly, the system contains no knowledge about its potential users. As a result that there is no differentiation between the output given to novice users, expert users and knowledge engineers. Because of this the system is incapable of any user-system dialogue. Since it can be argued that explaining is not a once and for all thing but depends on a process of negotiation any oracular pronouncement from the system will be fairly useless. Secondly, the system contains only shallow sources of knowledge unfounded in theory or first principles. This means that it is incapable of any answer to a question which goes beyond its shallow heuristics to first principles. This also means that the system has no real understanding of the domain it is trying to explain. It is a cardinal precept here that what is not understood can not be explained. Less importantly the system has no awareness that there are different sorts of explanation which may be appropriate in response to different questions.

Clancey's work on GUIDON (1979, 1986a) and NEOMYCIN (Clancey and Letsinger, 1981, Clancey 1985a) is an attempt to deal with some of these problems. GUIDON, a tutoring system, can be seen as a development of its predecessor, MYCIN, in at least the following ways: i) the system has (of necessity) a user-model and some user-machine interaction and ii) Clancey has begun to think about ways of organising the knowledge in the knowledge base (KB). NEOMYCIN is a consultation system whose KB is that of MYCIN reorganised to explicitly include strategic knowledge so that it can be used by GUIDON in teaching situations. The reorganisation and expansion of the MYCIN KB required for teaching purposes is also required for explanation purposes. Indeed it might be argued *a priori* that this would be so, given the close interdependence of these two activities. In particular, MYCIN rules had to be expanded to include knowledge such as the following. *Strategic knowledge* is represented in NEOMYCIN in terms of hierarchical meta-rules (Davis, 1980). This provides a vehicle for the explicit representation of problem-solving

strategies - in this case diagnosis of a disease. *Causal knowledge* is represented in terms of rules. *Disease process knowledge* is implemented in terms of frames whose slots are aspects of the disease process.

The application of these extensions to explanation is made clear in Clancey (1983). In this paper the work on NEOMYCIN and GUIDON receives a clear exposition based on a strong epistemological theory of how knowledge should be organised in the KB of an expert system. Basically, Clancey argues that a KB should be viewed in terms of three kinds of knowledge, apart from the empirical associations used in inferences:

- *support knowledge* i.e. the causal and disease process knowledge mentioned above.
- *structural knowledge* i.e. abstractions which are used to index the rules of the KB.
- *strategic knowledge* i.e. the organization of goals and hypotheses for problem-solving tasks such as diagnosis or interpretation.

One criticism of this work is that it is perhaps too tied to the MYCIN rule-based approach. Thus even though an epistemology is advanced which is generally applicable, the discussion is in terms of a particular approach. Clancey is too concerned with explanations which are explanations of rules. Little account is taken of the sorts of explanation which are appropriate in everyday life (i.e. not simply to do with rules) and in systems which seek to incorporate the causal and common-sense knowledge needed. More recent work by Clancey (Clancey, 1986) shows a greater awareness of the limitations of the rule-based approach and the need to incorporate models at various levels. Indeed in his most recent work, Clancey argues that expert systems should be seen as *qualitative models* of various domains.

Hasling, Clancey, and Rennels (1984) give a clear picture of how explanations are treated in NEOMYCIN and of the explanation of strategies. Explanation in NEOMYCIN is geared to the production of *strategic explanations* which are defined as an attempt to make clear the plans and methods used in reaching a goal (in this case, a medical diagnosis) (Hasling et al., 1984:5). Unlike previous approaches, NEOMYCIN produces its strategic explanations from an abstract, declarative representation of the strategy as a meta-rule. That is to say, that the meta-rules which implement these strategies are not domain specific to the domain of medical diagnosis. Thus, in NEOMYCIN, *Why* and *How* explanations are answered in

terms of the current task (for example 'test-hypothesis') being pursued as part of a general strategy such as 'group-and-differentiate'. Clancey (1985a) gives more details of the sorts of meta-rules employed in the system.

With NEOMYCIN we can see a progression towards a system with a more viable explanation capability. The NEOMYCIN KB contains different sorts of knowledge which are explicitly represented and clearly differentiated from each other. This has been done in terms of an epistemological theory of how a KB should be organised. In general GUIDON and NEOMYCIN represent a move away from the simple display of rules and rule traces. They are able to interact with the user and form relevant explanations based on multiple knowledge sources. However, no attempt has been made to enable the system to carry on a process of negotiation in order to determine a correct explanation.

3.2.2 The sociolinguistic approach

The work of Goguen, Linde and Weiner on the system BLAH has made use of sociolinguistic studies of human explaining behaviour to produce a theory of the structure of explanation. An explanation, viewed as a discourse structure, is modelled in the machine as a tree of statements and justifying reasons. The production of the explanation can be viewed as a series of transformations on this tree. This approach to explanation is also an approach to reasoning and hence argumentation as well.

BLAH - the theoretical background

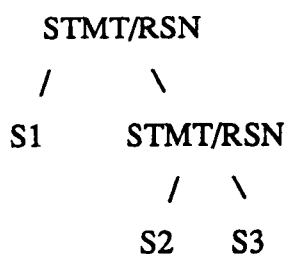
Goguen, Weiner and Linde have collected data from a number of psycholinguistic studies which has enabled them to produce their theory of the structure of explanation. It is clear from their paper (Goguen et al. 1983) that these authors do not see the explanation system as something which is attached almost as an after-thought to some reasoning system. They (ibid: 521) define reasoning as the 'internal mental processes' which result in reasoning 'expressed in verbal form'. They define the expression of reasoning to be explanation. Explanation is thus for them something which is central in the lives of human beings since it is the sole access we have to the reasoning of others. Given this it is also likely to be central in any interface seeking to be human-like.

Eschewing the normative approach of logicians and others they approach the study of explanation in an empirical manner by observing how people produce explanations

in commonly occurring situations. They hope that an understanding of how people explain will result in more comprehensible machine output.

Their theory is stated as follows (ibid: 523): 'We wish to regard the text as a sequence of transformations on an underlying tree structure which represents the abstract form of the argument being developed'. An explanation is viewed as 'a unit of language which purports to show why the speaker believes some particular statement, and (in most cases) is intended to cause the hearer to accept this statement'. As a tree structure plus transformations an explanation is amenable to computational expression. The authors make use of a representation of explanation parts as trees in which the leaves are statements and the nodes are labelled with the different kinds of subordination. The explanation is built up by a series of transformations on this tree. These transformations are signalled in the text of naturally occurring explanations by 'enunciants' (i.e. "any piece of text, of whatever size, which signals a transformation" (ibid pg. 537)

A typical tree might look like the following where S1-3 are actual propositions:



This tree can be read as saying *S1 because S2 because S3*.

BLAH - practical application

The theoretical work reported on in Goguen et al. (1983) has been implemented in the BLAH explanation system (Weiner, 1980). This is a knowledge based system where the assertions and justifications (statements and reasons) are produced by the rule-based reasoning component. The KB is also partitioned into system or user *Views* i.e. the assertions currently believed. The system works by outputting, from the reasoning component, a tree which represents a statement plus support (i.e. an explanation) which is structured in a natural way with subordinating parts as discussed above. The explanation generator translates this tree into the final output text. The tree is pruned by reasoning in the user's view to see what s/he knows (i.e.,

the partition of the knowledge base associated with the user is consulted). This provides an elegant means of circumventing the need for user models. The pruning which takes place as a result of this reasoning ensures that only relevant explanations are given. The explanation is also made more comprehensible by making it less complex.

One minor drawback of the system is the simplicity of the KB which it uses. Since the research is primarily concerned with explanation rather than reasoning this is understandable. Further, there is no reason to suppose that a BLAH-like system could not have a KB with multiple diverse knowledge sources.

This work has had a profound influence on my own work. I have attempted to follow the general approach in producing a model of argumentation which, while not being as mathematically elegant as the use of tree transformations by these researchers, at least provides a clear set of relationships between the various components. I have also adopted the use of the tree structure for my argument-1s. I have not made a systematic study of naturally occurring arguments but, as far as possible, I have used examples drawn from real arguments. An ASP might be seen as the attempt to produce a BLAH which can criticise a user's tree structure and sustain an argument about a topic as well as produce its own argument-1s.

3.2.3 The critiquing approach

There are two bodies of research which exemplify this approach - the work of Langlotz and Shortliffe on ONCOCIN (1983) and of Miller on ATTENDING (1983). In this approach the relevance of the explanation to the user's wishes is guaranteed since the program proceeds by asking for a user's plan (e.g. for therapy management) and then compares it with its own while producing explanations of the differences. By means of this mechanism, the attempt is made to return the control of the diagnostic or other task to the hands of the physician dealing with the particular case. The machine becomes an assistant rather than directive and intrusive. This aspect of the critiquing approach makes it particularly attractive for work in the humanities.

ATTENDING

The approach in ATTENDING is very similar to that of ONCOCIN, albeit without the interaction after the critique is given. The following quote from Miller (1983: 452) gives a clear resumé of the system's actions.

First, a *conversion routine* transforms the physician's plan, expressed as "menu

selections", into a tree-structured form called the physician approach tree (PAT). The PAT is then passed to the ADN [medical knowledge is stored in the form of augmented decision networks] *analyzer* which uses the augmented decision networks, together with the risks and benefits which augment their arcs, to analyze the proposed approach. This analysis produces the alternative approach tree (AAT), which includes the proposed approach plus any alternatives which ATTENDING has found. Finally, the AAT is input to the *prose generator* which produces the prose analysis.

In many ways an arguing system will be analogous to the above. Step one in such a system will involve the parsing of the user input and the creation of a tree like structure for the user's argument as well as the integration of this into the overall argument. Step two will involve the finding of a suitable response. Step three will involve the output of the response to the user's argument by means of text or a graphical display. The most important point from an argument viewpoint is that while selection of an alternative argument is analogous to argument response, ATTENDING makes no attempt to assess the alternatives. This is something which is probably going to be important in any argument since at the very least assessment is needed to decide on the point in an opponent's argument to attack next.

ONCOCIN

Langlotz and Shortliffe (1983) discovered that the routine giving of advice by ONCOCIN was standing in the way of physician acceptance of the program. This was one of the main aims in the system. Thus, although the physician could override ONCOCIN's recommendation, frequent such overridings became annoying to the users. Langlotz and Shortliffe decided that this barrier to acceptance could be overcome if the system, instead of routinely offering advice, was modified to criticize the therapy plans of the doctor using the system. This critiquing approach is a development of work done by Miller mentioned above. The ONCOCIN system is intended to make such critiques more acceptable to the user by providing explanations of the *differences* between the plan proposed by the physician and that recommended by ONCOCIN. The critiquing approach thus allows the doctor a greater say in the treatment plan. Furthermore the system provides an opportunity for educational use, in that the user can be provided with explanations of the differences between his or her plan and that of the system.

Comparison is done by the method of *hierarchical plan analysis*. In short, this is a technique based on the fact that decisions about which kinds of treatment should be selected depend on other decisions at a different level. For instance, decisions about radiation therapy may be dependent on decisions about chemotherapy. In the comparison process the system determines whether there are significant differences between matching components in the two plans. Explanations are generated for each of these differences with the explanation dialogue focussed accordingly.

Again, given the limited aims of the system builders, it is not surprising that the critiquing approach does not *on its own* represent a solution to the problems of explanation. I hope, however, that it is obvious from my general remarks earlier and from my comments above that I feel that adequate explanations will not be forthcoming from an expert system until this includes elements from a BLAH-like system and from the critiquing approach and that this is best represented by the following approach.

3.2.4 The cooperative approach

This is the approach adopted by such researchers as Rector, Newton and Marsden (Rector et al 1985), Kidd (1985), Worden et al. (1986) and Knight (1986). The approach is exemplified by a shift in emphasis from a model of expert systems as Delphic oracles toward a more cooperative model in which the system and the user act together to solve a problem. This may involve, as in the suggestions of Kidd, a system which is capable of a process of negotiation in order to produce the best explanation. Apart from the work on cooperative interfaces to expert systems there is also work on cooperative front ends to databases (Kaplan 1982) and research on computer support for cooperative work (e.g. Winograd and Flores 1986).

The basic reason for the introduction of the cooperative model is, as has been mentioned before, the extreme reluctance of users to accept expert systems. One possible means of overcoming this reluctance is to have a system which does not provide a once and for all answer but which 'cooperates' with the user in performing some task. In one sense the cooperative model can be seen as an extension of the critiquing model in that it is one function of an assistant to criticise the decisions of the person being assisted. A full assistant will need to perform other functions as well. For instance the assistant needs to know when to take the initiative, what to do when a system decision is overridden, explain its own decisions and criticise the user's when

asked.

These ideas have been implemented in a system for Electromyography examinations which are aimed at diagnosing peripheral nerve disorders such as Motor Neurone Disease. In addition to the domain knowledge and knowledge about strategies, the EMG assistant has knowledge of how to be an assistant. The chief interest of the paper is in the alternative model of explanation it suggests - in which the user and system cooperate in a task, the system provides explanations of its decisions and expects explanations of the user's decisions. In order to accomplish this the system conducts a dialogue with the user. This dialogue would, in an cooperative explanation system, enable the system not only to determine the most appropriate explanation to give but to understand the explanation given by the user.

3.2.5 Conclusion of this review of explanation facilities

As a result of this review of explanation capabilities I contend: firstly, that there has been a trend towards greater user/system interaction, and secondly, that there has been a trend towards cooperative rather than oracular responses. An ASP is intended to be both interactive and cooperative and can thus be seen as a natural outgrowth from the above research.

There are however differences between the work of Wordern et al., Knight and Rector et al. and myself. For instance Rector et al. envisage a cooperative expert system as ideally one which has two parts - the 'problem representation system' (PRS) and the 'assistant system'. It is the job of the latter to cooperate with the user, with communication via the blackboard-like PRS, in finding the solution to a problem. My system is ultimately designed to bring about agreement rather than cooperation. Of course, as Grice has shown, no form of conversation is possible without cooperation at some level. However my system tries to cooperate in the user's overall goal (of reaching the truth of some matter) by being as argumentative as possible. The user's arguments are put to as extreme a test as the system can possibly devise and argument is sustained for as long as possible. Thus neither the system or the user is the master. There is a game played in which truth is the prize.¹ The system plays the role of death in a chess-game with the user in which truth and reality are the prizes rather than the user's life. That this is so is a natural

¹ This is the notion of argument or *dialectic* to be found in Plato's dialogues. According to Plato we use dialectic as a means of experiencing the transcendent world of the forms.

result of the battle-metaphor which lies at the heart of our notion of argument. This in turn is related to Popper's notion that what we are about in science is finding ways of falsifying hypotheses (1963). Thus argument is not to be seen in a negative way as a war in which power relations are fixed but rather as a series of duels between equals in which some matter of honour is settled.

In the above I have, of course, left out many approaches to explanation in expert systems. This partly for reasons of space but also because the above are the most relevant to the work I am reporting in this thesis. Apart from the above perhaps the most important contributions are:

- The computational approach to explanation (Swartout 1983, Neches et al 1985) in which automatic programming techniques are used to produce an expert system, leaving a trace of the refinement structure which can subsequently be used by the explanation capability in justifying the program's code. This work is based on the assumption that the reasoning which system designers go through in producing the system is what is needed in producing explanations. The automatic program generator makes use of domain models and domain principles (problem-solving strategies) in creating the system.
- The graphical approach in which explanations are displayed as mixed text and graphics using current work station technology. Representative work includes that of Clancey on Guidon-Watch (Richer and Clancey 1985) and Langlotz et al (1983) on the interface to the ONCOCIN program discussed above. In the latter case the interface is composed of a screen version of the form used for recording the experimental plans for treating cancer with chemotherapy. The principal aim being ease and naturalness of use. As we shall see below (in the discussion of hypertext) similar developments are being made in displaying and interacting with other kinds of data and knowledge bases.

Other approaches to explanation in expert systems can be gleaned from the proceedings of the Alvey sponsored workshops on explanation which have been held over the past few years.

3.3 Approaches to argumentation

Unlike the research in explanation presented above, in argumentation there is neither a) a clear picture of the sorts of research carried out; nor b) a coherent and organic progression from one area to another. This is partly because a lot of the research in explanation is based on the work done on Mycin thus giving it a feeling of connectedness. On the other hand I feel that the main problem is that the area of argumentation - a battleground for centuries - is one which does not divided clearly into different approaches. Because of this the work I shall present is not split by area but by researcher or research group. I will present the work of four groups or individuals and one more diverse set of approaches:

Yale group

McGuigan and Black

Cohen

Reichman-Adar

Hypertext and similar approaches

3.3.1 The Yale group - the process model of argument

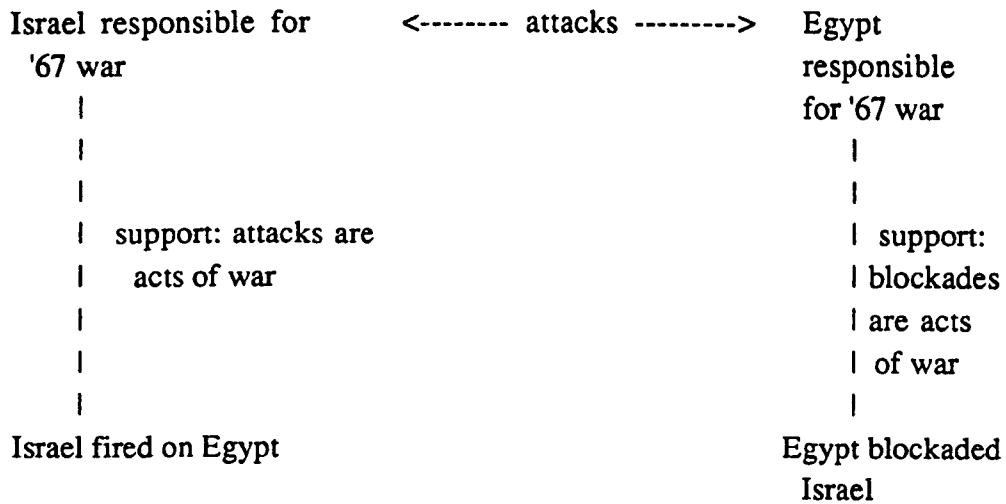
The above is a convenient *nom de guerre* for a group of researchers at Yale who are involved in research into various aspects of argumentation. While their concerns are not identical - there are at least two main areas of research - they are rendered fairly homogeneous by their background in Schankian theories of language and memory structures. The two main areas of research are i) adversary arguments ii) the understanding of arguments in newspaper editorials.

Adversary arguments

This is represented by the work of Flowers, McGuire and Birnbaum (1982), Birnbaum (1982) and Flowers and Dyer (1984). As presented in the former paper the authors are seeking to formulate a theory of the processes which are needed in comprehending and generating argument steps. They have implemented their theories in a computer program (ABDUL/ILANA) which can take either side in an adversary argument about the Arab Israeli war of 1967. Flowers et al. make a strong distinction between *adversary* arguments and *persuasion* arguments. The goal of the former is winning, of the latter agreement. By and large the sorts of argument I am concerned with are

the latter - argument used as a means of cooperative problem solving. However I feel that such arguments (if they are to deserve the name) must be at least as well argued as adversarial arguments. Flowers et al. make the distinction in terms of the sorts of moves allowable in the former which are not made in the latter. In particular an adversarial argument allows for personal attacks. This may be largely true. However the sorts of attacks they describe are not really personal attacks as such but attacks on ingrained positions. There is thus not so much of a distinction here. The cooperative arguer can attack the background beliefs of the user. Neither kind of argument easily countenances the sorts of degenerate arguments or rows in which personal abuse is allowed. That this is so is shown by the example of legal argument. This is essentially adversarial but has as its goal the solution of a problem i.e. the doing of justice. Thus what Flowers et al. have to say is of relevance to the cooperative arguer.

In order to implement the process of arguing Flowers et al. have made use of the fact that arguments have a certain structure. In their terms they are made up of propositions which are connected by *argument relations* either of *attack* or *support*. The possibility of successfully engaging in an argument depends on *argument tactics*. For instance in order to attack an argument one can either: 1. attack its claim; 2. attack the evidence offered in support of the claim (i.e., its grounds); or 3. show that the evidence fails to support the claim. In order for the arguer to relate argument steps to the argument as a whole an argument must be stored. Flowers et al. use an *argument graph* - 'a network of propositions connected by argument relations'. They go on (ibid: 281): 'This graph is important both for understanding and rebutting in an argument, because it constitutes the conversational context into which a new utterance is integrated, and found to be coherent.'



An example argument graph (from Flowers and Dyer, 1984).

The notion of the argument graph was very influential in my own model. Thus the graph given above is very similar to the examples given in the introductory chapter. The notion of (external) relations of attack or support between the two arguments is retained. The principal difference is that the notion of (internal) support between evidence and claims exemplified above is replaced by the notion of a warrant which relates grounds and claims. (A notion introduced in the later work of this group.) This avoids confusion between the support relation between two arguments and the support relation between grounds and claims.

Components of an argument graph which are likely to recur are called *argument molecules*. The importance of these is twofold: a) they facilitate argument understanding since they can act as components of the argument graph, and b) they restrict the range of potential responses by making clear which propositions are most vulnerable. It is not entirely clear from the paper what kind of restriction is involved here. I take it that they can be used to determine which of the propositions making up an argument are more or less sustainable/attackable. This is made clearer in the Birnbaum paper where he shows how such structures could be useful in finding the propositions which are 'likely candidates' for attack. Birnbaum (1982: 65): 'In essence, a molecule packages knowledge about the logical structure of an argument fragment in a way which makes explicit which potential responses would have some logical force, and which would not'.

Flowers et al. also attempt to show that possible rebuttals can be found in the comprehension process. This comes about because, in performing a search of the causal chains which they use to represent historical facts, the system can note 'near misses'. It seems intuitively that this is something like what happens when we argue - possible rebuttals are accessed as a new argument is presented to us. I am not sure however that the ability to use near misses in this way is anything more than an artifact of the sort of representation they use. If, for example, memory is represented by production rules and remembering as the process of firing these rules then it is not so easy to see how 'near misses' could be determined. Perhaps all this shows is that the Schankian type representation is superior. Note that this is related to the Yale group's belief that argument is not an explicitly planned activity but is essentially *opportunistic* in that rebuttals are often found while engaged in doing something else (Birnbaum, *ibid*: 63). McGuire, Birnbaum and Flowers (1981) make the point that it is only when such reminders are not found in this opportunistic manner that the argument graph is used for the generation of responses. As McGuire et al. conclude: 'It seems possible that a theory of conversation (or more specifically, of argumentation) based on this kind of opportunistic processing can reconcile our everyday perceptions of conversations (or arguments) as being, on the one hand, planful, and on the other, wandering and disorganized.'

One further criticism of the above work, raised by Cohen (1987), is that while it deals with what responses to generate it has little to say about how participants structure arguments and not much on how arguments may be assessed.

Understanding newspaper editorials

The two papers I shall concentrate on here are Alvarado, Dyer and Flowers (1985) and Alvarado, Dyer and Flowers (1986). This work represents a development of the work discussed above and an application of it to a particular kind of text - the newspaper editorial. Again because of its concern with natural language issues this work is only of partial relevance to my own. In fact it is of more relevance to a possible development of my work where the arguing system is used in comprehending the arguments used in, say, literary critical texts. The work presented in these papers is also only concerned with argument *comprehension*.

Alvarado et al. call their model a *conceptual* model in contrast to the *structural* model of Cohen (see below). Their system understands the arguments in newspaper

editorials not merely by recognising the structural relations between parts of the argument but by building a *conceptual graph* which '...captures interactions between goals, plans, events, states, beliefs and argument units'. One new aspect is the inclusion of Toulmin's warrants as more basic beliefs relating conclusions and evidence.

There are two major new features of the Yale Group's work: argument units and reasoning scripts.

- Argument units

According to the 1985 paper, '*Argument units* (AUs) are abstract reasoning structures which organize: (a) belief, goals, and plans; and (b) support and attack chains of reasoning and relationships in arguments'. While argument molecules are not mentioned in these papers, I think that it is safe to say that units and molecules are related notions. Alvarado et al. (1985) mentions seven AUs: AU-ACTUAL-CAUSE, AU-OPPOSITE-EFFECT, AU-EXPECTATION-FAILURE, AU-HYPOCRISY, AU-ACTUAL-EFFECT, AU-EQUIVALENCE and AU-RELEVANT-ISSUE. These can be illustrated by AU-OPPOSITE-EFFECT which contains the following reasoning chain:

Although Y believes that executing his PLAN P will achieve GOAL G, SELF does not believe this because SELF believes that executing P will thwart G. Therefore, SELF believes that P is bad.

where SELF is the originator of the AU. As we can see, unlike the situations discussed above, the AU captures reasoning chains which are essentially to do with the goals and plans of the participants. For Alvarado et al. (1985):

...following an argument involves recognizing these constructs [argument connectives such as 'but', goal, plan, belief relationships], accessing the specific conceptualizations they refer to, mapping from them into their appropriate argument unit, and triggering that argument unit's inference rules for recognizing belief, support and attack chains of reasoning and relationships.

As they conclude 'We believe that all arguments are composed of configurations of a fixed number of abstract *argument units*'. An AU organizes several support or attack relationships and provides the means for integration into the argument graph as well as a means of accessing the graph. There is also a taxonomy of AUs depending on whether the support/attack relationships are to do with (a) the implementation of plans (b) the pursuit of goals or (c) the holding of beliefs in ideological contexts.

- Reasoning scripts

In Alvarado 1986 it is made clear that they think of their theory as a theory of reasoning as well as argument. As far as I can tell reasoning scripts are used as a means of identifying and tracking chains of reasoning which support beliefs. Thus they are necessary in building the conceptual graph. I assume also that they will be used in concert with AUs to fill in the justifications in these (higher level) reasoning chains. At any rate they are a development of a notion mentioned in Flowers et al. (1984: 658). Here 'A fundamental notion ... is that human reasoning frequently makes use of previous examples, prior chains of reasoning formed from previous arguments, and adaptations of situations related to the current problem'. The idea is that 'general rules' are used in the first instance to produce an elaborate chain. This is then stored as a script which is then accessed in solving similar problems. In Alvarado et al. (1986: 251) the reasoning scripts are more closely related to the plans and goals of the participants.

In OpEd [Alvarado's program], *reasoning scripts* are used to organize prespecified reasoning chains involving cause-effect relationships among politico-economic goals, plans, events, and states...OpEd recognizes and instantiates these reasoning scripts when following belief justifications which contain structural gaps...

To illustrate this the following is the the \$R-DROP-FOREIGN-SPENDING--> DROP-JOBS script:

IF COUNTRY C1 spends less on PRODUCT P produced by PRODUCER P1 from COUNTRY C2, THEN there is a decrease on the EARNINGS of PRODUCER P1. AND IF there is a decrease on the EARNINGS of PRODUCER P1, THEN there is a decrease in the number of OCCUPATIONS in PRODUCER P1.

One criticism highlighted here but true of the whole approach is that the rule mentioned here is very specific to reasoning in a particular domain - economics. The authors claim that their approach is generalizable to any editorial. That may be so (since they are usually concerned with political or social or economic issues) but I doubt whether it can so easily be transferred to other arguments.

3.3.2 McGuigan and Black - the rhetorical model of argument

Another perspective on the study of argument is provided by McGuigan and Black (1986). This approach, while not as fully worked out as that of the Yale group has the

advantage of support from empirical studies as well as relative independence from the ideas of Schank on memory and language comprehension.

In a manner similar to rhetorical studies of argumentation they have produced a typology of argument which is very useful as a starting point for my own work. As a result of an examination of a number of arguments, McGuigan and Black have divided argument structures into three types: argument by analogy, categorical argument and causal argument. Categorical argument (or argument using categories) includes deduction. What is interesting about this division is that they believe '...that the representations and processes used for argument also serve for understanding the world in general'. That is to say that they believe that the process of argument comprehension (and generation) are not distinct from the process of understanding generally. They support this by showing: a) that categories are important both in argument and in the process of gaining understand about the world; b) that the physical world cannot be comprehended without an awareness of causal relations; and, c) analogies are an important means of grasping the significance of new situations and assimilating new knowledge. As a result of this, their argument comprehension program proceeds in a fashion more or less indistinguishable from the way a general world understander would. While I agree that argument understanding is related to world understanding (arguments have an epistemic role) I feel that they have not proven that there are not specific knowledge structures for argument. Apart from this their approach can be criticised since they have not clearly distinguished between argument comprehension and generation or between comprehension and assessment. The latter distinction is necessary since it is possible to comprehend an argument which is assessed as being weak.

McGuigan and Black have implemented their ideas in a program called MAGAC. As it is presented in the paper, this program deals with arguments (comprehends and assesses them) by instantiating frame like structures called support and conclusion organization frames (SOFs and COFs). The program proceeds by utilizing background knowledge to fill the empty slots of the COF. As far as I can tell this knowledge is placed in the system by the builder - there is no means of inferring it during processing.

While the work of McGuigan and Black contains some good ideas (on strategies for example) and has been evaluated by empirical tests I feel that it fails on one major count: There is no clear and precise theoretical backing for the implementation. It is

simply asserted that argument comprehension is accomplished by the same structures as world understanding - there is no theory to link these. Nor is there any theory or model to link comprehension, evaluation and production.

3.3.3 Cohen - the structural model of argument

Cohen (1987) is principally concerned with understanding natural language (as anyone interested in arguments must ultimately be). However the distinctions she draws and the algorithms she presents are of relevance to any work on understanding and evaluating arguments. Cohen's paper is divided into three main sections. These deal with the three main theories she advances as part of the background theory from which a computational model of argument might be derived. These are:

- a theory of expected coherent structure - i.e. a specification of the kinds of proposition and their ordering;
- a theory of linguistic clue interpretation - Cohen, since she is dealing with natural language, develops a taxonomy of possible clue words or words (especially connectives) which can give information about the user's intention in a proposition;
- a theory of evidence relationships i.e. what counts as evidence for what.

For Cohen an argument is a set of propositions. These can be analyzed one at a time to see where they fit into the argument as a whole. The argument is represented as a tree. The interpretation of an argument can be construed as the assignment of a place in the tree for each new proposition. Its position depends on a set of reception algorithms. Its relationship to other propositions (if it is related) is decided by an 'evidence oracle'. As far as I can see the new proposition is fitted into the tree only if it is passed by the evidence oracle. This may be a drawback since in certain categories of argument, for example those with many sub-arguments, it is necessary to assess whether or not a proposition detracts from the argument. In this case it needs to be stored as part of the overall argument even though it fails.

The evidence oracle is perhaps the most interesting part of the system - especially since none of the other researchers have devoted much attention to it. The basic definition of evidence is given as:

A proposition P is evidence for a proposition Q if there is some logical connection from P to Q - i.e. some rule of inference such that P is premise to Q's conclusion. On the face of it this seem to equate arguments with deductively valid arguments.

However, Cohen takes account of the fact that we are not deduction machines. She introduces two notions to deal with this - 'modus brevis' and relaxed logic- as well as a notion of plausibility which includes reference to the possible beliefs of others. The oracle works a 'black box' which accepts a set of two propositions and determines whether there is an evidence relationship between them. To do this it must: a) identify missing premise(s) and b) verify the plausibility of the missing premise(s) In order to achieve b) the system must:

- use logic in the system knowledge base
- relax logic in the system knowledge base
- stereotype speaker i.e. form a user knowledge base
- judge plausibility i.e. test against the hypothetical beliefs of an ideal third person.

Modus Brevis is a notion developed in a paper by Sadock (1977). The rationale behind his work is that in arguing we rarely present fully valid deductive arguments. He argues that we employ deduction in arguments but that these are often presented in a truncated form i.e. if we think of an argument as a deductive syllogism, that we miss out the major or minor premises. Thus for example in the argument presented by a below:

a: The Falklands war won Thatcher the election.

b: Why?

a: Because she was seen as an 'iron lady'.

the missing major premise is something like 'All iron ladies win elections'. As Cohen says 'The common form for arguments, then, is one where the hearer must supply missing statements in order to establish the connections for the representation of the argument'.

Relaxed logic includes what other writers have called induction and abduction. If the system fails to find a logical connection then it tests to see if the statement can be derived by relaxing the requirements for deduction. For Cohen relaxed logic is a means by which the system fills in either a user knowledge base with plausible premises (i.e. those it is plausible for the user to believe) or its own knowledge base with the results of this plausible reasoning or with generalizations. Abduction is more clearly used when the system does not believe a premise and has no prior knowledge of a particular speaker (or stereotypical knowledge of a class of speakers) then the

system must postulate new facts. Cohen speaks of this as reasoning from the beliefs of a hypothetical person. One means of doing this would be an abduction from $P \rightarrow Q$, Q , therefore P , where P is not believed by the system but could plausibly be held by the user.

In conclusion then, I feel that there is much of value in Cohen's approach if only because she allots sufficient concern to evaluation in the comprehension of arguments. As with the work of the Yale group it is clear that argument, while it is a social interaction on one level, is deeply concerned with issues to do with valid and plausible forms of inference.

3.3.4 Reichman - the discourse model of argument

It is very difficult to summarize Reichman's (Reichman-Adar, 1984, Reichman, 1985) work in a short space. However the main points of her theory are:

- a) a conversation can be viewed as a sequence of moves - e.g 'support' and 'challenge' moves are particularly relevant to argument;
- b) the conversation has an underlying structure which enables us to follow its many 'twists and turns';
- c) the structure is a hierarchy of discourse units (or elements which play a particular role in the conversation);
- d) at any one time there are two context spaces (the name she gives to discourse units embodied in conversation moves or series of utterances) active in a current section of conversation - the *ground* or preceding and controlling context space and the *figure* or currently being developed space;
- e) context spaces have an internal structure which includes the set of utterances in the space, a pointer to its controlling space and a specification of the type of relation to the controlling space;
- f) the grammar which controls movement between context spaces can be encoded in an Augmented Transition Network (ATN).

In short then Reichman envisages a conversation as a series of discourse moves or conversation moves. As Grice (1975) pointed out, these moves can be more or less appropriate given the current state of the conversation. Reichman's principal goal is to render Grice's maxims for conversation more computationally tractable. To this end there are rules for determining when a conversational move is appropriate. In the

model which she provides, conversational moves are represented as the acts in an ATN-like grammar. As test conditions are met, using information about the conversation stored in the context space (CS), the grammar gives the moves which are currently possible. The CSs thus constitute the static information about the conversation while the ATN represents the dynamic aspect of the conversation. In this way a context space 'fulfills' the conversation move. Reichman means by this that the new context space is set up as the static record of the move. To achieve this the context space contains slots which organize the propositions which make up the move, representations of the components which are necessary for a move to be 'fulfilled', pointers to other context spaces and the status of the CS in the conversation (i.e. active, controlling and so on). Reichman mentions two types of context space: issue and non-issue. Issue is further divided into epistemic, evaluative and deontic CSs. All of these can be debative or non-debative (if the latter they will have slots for protagonists, antagonists, counterclaims and countersupports). Non-issue CSs include comment, narrative and support.

I have two main difficulties with Reichman's work. While it provides a feasible means of delineating the legal moves in a conversation and of dealing with the thorny problem of Grice's maxim of Relation (relevance is relative to the context spaces currently operative), it is, firstly, too constraining, in that natural conversations are more creative, and, secondly, somewhat deficient in giving actual strategies for assessment and next-move generation. In everyday arguments we are aware of when we are losing an argument or, looking back, of who the winners and losers were. Thus we must have some means of assessing the overall strength of an argument. This is of vital importance in an automatic system. The system will only be able to tell that it has lost the argument if there is some explicit means for evaluation. This awareness is important since the system needs to be convinced of this defeat before it will change its mind. Thus arbitrary changes to the knowledge base will be avoided and a principled and human-like means of knowledge acquisition can be implemented.

3.3.5 Hypertext and argumentation

While there are many examples of the use of hypertext systems as a means of storing, retrieving and displaying arguments, I will concentrate on two: a) VanLehn b) Marshall. For a general resume of hypertext systems the reader should consult

Conklin (1987a,b). Other systems which deal with argument include Trigg (1986) and Begeman and Conklin (1988). There are yet other systems which, while not really hypertext systems, have some similarities. These include the work of Lowe (1985), Stefik and others on Colab (Stefik, Foster, Bobrow, Kahn, Lanning and Suchman, 1987) as well as the general area of what has become known as 'groupware' (Byte special report December 1988). As a third example in this section I will briefly discuss the work on Colab.

- Van Lehn (1985) discusses the use of NoteCards as a means of representing the argumentation necessary in theorizing about the learning process involved in learning procedures. As he says 'The main job of argumentation is to contrast the explanatory power of various sets of hypotheses. Metaphorically speaking, the theorist takes several sets of hypotheses and sees how many bugs each can explain.' (1985:4) Basically Van Lehn took the argument structure of his thesis and represented it in NoteCards format. A Notecards 'card' or window contains each hypothesis in the theory and has pointers to other cards giving the arguments for and against the hypothesis. As the report suggests it is relatively simple to take a linearly organized document as a tree of arguments and translate this into the network available in Notecards. Indeed it is one feature of such systems that they allow the fairly arbitrary representation of such structures. Unexpectedly VanLehn found that in two instances the use of NoteCards forced theory reform. Firstly when adding new hypotheses VanLehn found it easier to represent what had been a tree as a matrix (see Figure 3.1). This resulted in a table with clear representation of whether a piece of evidence supported a hypothesis or not. When this was done a major re-evaluation of the strength of the differing hypotheses was achieved. 'In short, sloppy reasoning, abetted by a poor rhetorical organization, allowed the suppression of a winning hypothesis.' (1985:6). The second discovery was made when NoteCards was used to display the graph made up of the cards and the pointers. When this was done it was discovered that relationships between the various issues (about which hypotheses were made) were missing. As VanLehn points out this should not be the case in a properly structured theory. In effect what had been left out were the decompositions of the large issues into smaller ones. These decompositions were not made explicit in the original document and therefore represented unexamined and unexaminable assumptions. In the event VanLehn discovered a new decomposition for one of the

issues and hence a new version of his theory. As he concludes (1985:9):

However, there is no cadre of professional argument-organizers who are the analogs of printers and graphics designers. Nor can there be such a cadre. The organization of arguments is too strongly coupled to their content. ... To put it differently, argument has an organization whether one likes it or not. ... Someone has to give the arguments a good organization. And that someone must be the thinker who is responsible for the argument's content. ... NoteCards' organizations are *emergent properties of the content of the arguments*...NoteCards' emphasis on emergent organizations makes it harder to fool oneself. Perhaps the hardest job that a theorist has is to discover the assumptions that he or she is making....NoteCards seems to provide such an aid. ... The NoteCards database is about as close as any written artifact can get to expressing a whole theory. NoteCards is a first step on the path towards a theoretician's workbench that is a synergistic combination of human and artificial intelligence.

- Marshall (1987) discusses, among other applications, the use of NoteCards as a means of representing 'the logical structure of an argument' (1987:253). In this case the card and pointer mechanism is used to capture the structure of arguments in a manner based on that of Toulmin (1958). (I will have more to say about Toulmin's work in chapter 4.) In this case the domain is that of legal arguments. The structure used to represent these can be represented as in Figure 3.2 and is one which I have adopted for my own display of argument-Os. Using this graphic representation, the relations between the different elements in the argument can be shown and one argument can be linked to other arguments. This facilitates the comprehension of the argument.

Marshall raises the important issue of the level at which argumentation structures should be represented and the semantics of the representation. As she argues a hypertext system allows a range of representations at different levels of the the same argument. Thus an argument can be seen as a whole. Alternatively the system can allow a process of 'zooming' to finer and finer details of the argument (cf TPM, the Prolog debugging tool discussed by Eisenstadt and Brayshaw, 1987). At the lowest level these would be the individual propositions which make up the argument.

Marshall also commends the use of matrix or tabular representations of the argument (similar to those of VanLehn) as a means of presenting an assessment or as a means of displaying the relations of support and/or attack between propositions.

To anticipate some points to be made in the final chapter of this thesis, Marshall

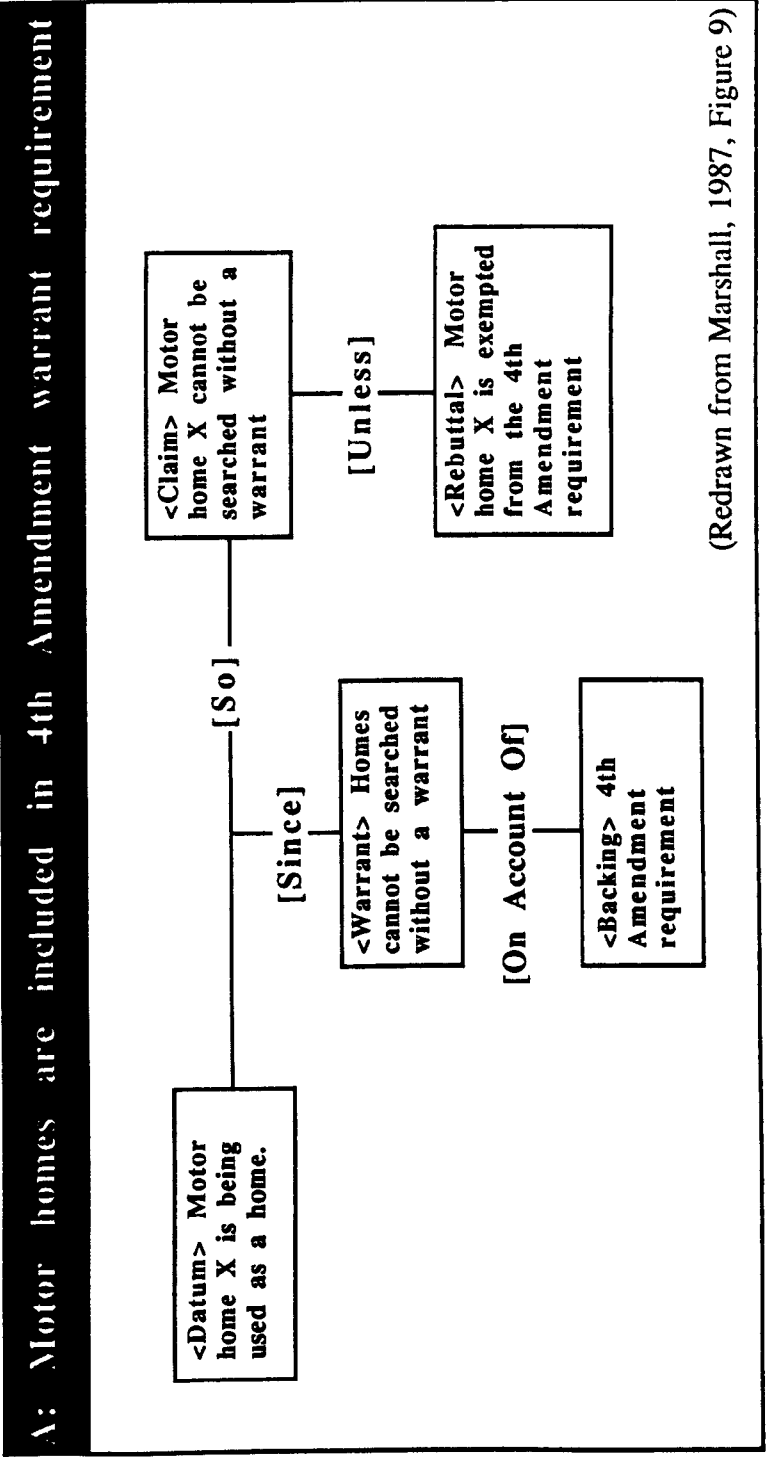


Figure 3.2 - Marshall's graphical Toulmin structure

points out (1987:268) that:

[One] way hypertext systems can be extended is by facilitating programmatic interaction with user-created structures and substances. For example, if a representation is used by an expert system or some other type of inference engine, it may be necessary for the program to interact with the content of a node or object to extract the knowledge inside it. Since hypertext is a good vehicle for capturing and structuring knowledge, programmatic interpretation of a hypertext network appears to be a logical extension.

This is a point also made by VanLehn and again more forcefully by Halasz (1988).

'The integration of hypermedia and AI technology is an interesting direction to explore.

In many ways, hypermedia and knowledge-based systems are a natural fit.' (1988:

847) As I shall suggest below (chapter 8) argumentation is a natural means of integrating these two technologies.

- The Colab project as reported in Stefik, Foster, Bobrow, Kahn, Lanning and Suchman (1987) is an attempt to create an environment which can be used to facilitate 'collaborative problem solving'. To do this each user has a workstation connected to other workstations by a local area network so that each user can see on his or her screen the same public information that the others see. Users can interact verbally or by means of software which allows them to draw or write on the shared screen areas. Two pieces of software are of especial interest: (a) the Cognoter and (b) the Argnoter.

The former, the Cognoter, is used to facilitate the production of a communally produced presentation. To some extent this program is a communal version of other outliners commonly packaged with word processing software. However the Cognoter can be used in all three phases of the production of an outline: brainstorming, organizing and evaluation. In the brainstorming phase concepts are entered freely on the screen. In the organization phase links are drawn between concepts to represent their order and ideas are grouped together. In the final phase, evaluation, the participants reorganize, amend and delete parts of the outline. As Stefik et al. point out (ibid: 36), this tool can itself be used to facilitate human-human argument exchanges:

In Cognoter, the various decision-making processes are separate and distinct operations. Delaying deletion until the last phase, for example, provides a more visible basis for argument in the sense that an argument for deleting an idea because it is not relevant may be more convincing when that idea is not visibly

linked with any others; or arguing the unimportance of an idea may be more convincing when the competing ideas are available for comparison.

However in the second tool, the Argnoter, work is proceeding to produce a tool which will act as a sort of argumentation *spreadsheet*. This tool imitates the arrangements of meetings aimed at trying out proposals which the Colab team have used without computer aid. The goal is to produce the best proposal. Again the tool has three phases: proposing, arguing and evaluating. In the proposing phase each proposal is given a description (which may be text alone or include graphics) and a name. In the arguing phase reasons for and against the various proposals are written beneath the appropriate proposal and distinguished as being either for or against the proposal. In the evaluation phase the assumptions behind the arguments for and against proposals are requested. As Stefik et al (ibid: 39) put it 'Assumptions, in Argnoter are expressed as statements about statements'. *Belief sets*, or sets of statements accepted as valid or rejected as invalid by the participants are used by the Argnoter to evaluate the proposals. As Stefik et al. explain (ibid: 39):

The proposal display is generated by stepping through the arguments about the proposal, looking up the assumptions, and then displaying those arguments that are supported in the specified belief set. Multiple belief sets may coexist, and any participant is able to create...belief sets. The belief sets are intended to characterize different generic points of view.

Evaluation proceeds by selecting and ranking the various evaluation criteria such as feasibility and cost which the participants want to use. Thus proposals can be viewed in relation to different criteria as well as in relation to different belief sets. The best proposal is therefore the best in relation to a particular belief set and as assessed in terms of a specific set and ranking of criteria. The assessment does not seem to be automatically performed. The system simply displays the proposals in the light of the beliefs and criteria and the decision is left up to the meeting.

This system has many attractions. For instance it can deal with many domains since it is only concerned with the structure of the argument not its content. As Stefik et al. point out (ibid: 39):

Argnoter need not understand the meaning of design proposals: It need only differentiate between proposals, arguments, assumptions, and belief sets, and compute the relevant logical support relationships.

While I am impressed by this work I have three main criticisms:

- (i) It seems unlikely that any real evaluation can be done without an analysis of the semantics of what is being evaluated. Much is made explicit in Argnoter but apparently not the semantics of the propositions. The system relies on a shared semantics on the part of its users.
- (ii) In any real argument the debate can be followed in many dimensions. Thus there needs to be a means of allowing debate about the belief sets before evaluation can occur.
- (iii) There is more to a viewpoint than a set of beliefs. While the use of belief sets goes some way towards the modelling of these there will often be a structure in these i.e. a theory or a model of a situation in which certain beliefs depend on others and so on. This constitutes another possible dimension of the argument.

Nonetheless there is much of interest in the Colab project. If only as they say (ibid: 40) because:

...making the structure of arguments explicit facilitates consensus by reducing disagreement that arises from uncommunicated differences.

3.3.6 Conclusion of this review of computational models of argument

If the relation of ASPs to explanation capabilities is one of continuity, the relation to the models discussed above is more complex. ASPs make use of elements from almost all of the models discussed. These are summarized in the next section. They also fulfill many of the same functions. In particular:

- ASPs (like the Yale group and Reichman) attempt to respond to user arguments;
- ASPs (like Cohen's model) check user arguments;
- ASPs (like the hypertext models) make use of graphical displays of arguments;
- ASPs (like McGuigan and Black's MAGAC) contain a typology of arguments.

At the same time ASPs can be used to provide an interface to an expert system which subsumes explanation capabilities. In this, and in its integration of all these elements, the ASP is substantially different from any of these approaches.

3.4 Conclusion

By way of a conclusion I would like to outline a series of lessons which have been imparted by the examples of the systems discussed above. In the next chapter I will develop a model in the light of these lessons.

Lessons from the research on explanation

- all aspects of the domain knowledge (including structure and strategies) must be represented explicitly
- this knowledge should be separated
- particular knowledge representations are appropriate for explanation structures (especially tree-like structures in which a statement is supported by reasons which may in turn be supported by reasons)
- in order to deal with explanations the knowledge base should be divided into views
- believable representations need the results of empirical research into human explanations
- explanations are open-ended
- explanations are conducted within on-going conversational exchanges
- *an* explanation is a structure; explanation is an activity which produces this structure
- it is possible to provide a graphical visualization of an explanation

Lessons from the research on argumentation

- data structures are needed for representing arguments in terms of individual steps and overall arguments
- the processing of arguments can be thought of in terms of parsing, checking and response
- structures for representing strategies for argument production and generation should be available
- a typology of argument types is needed
- there should be a computational means of dealing with these types
- (from hypertext) it is possible to have multiple representations/views of an argument both textual and graphical and a means of interacting with these/ traversing them

-o0o-

Chapter 4 - Models of interpretation and argument

Argument: Interpretation may be viewed as a process of applying transformative rules to facts and the results of other transformations in order to produce a set of claims and supporting reasons (an argument-1). Argumentation may be viewed in terms of arguers, arguments and domains. The notion of argument style is a means of determining which types of reasoning are operative in a domain. Relevance concerns the fact and warrant tokens.

In the previous chapter we presented a historical account of the background of the ASParch project and suggested that explanation and argument are closely related. In the present chapter I put forward the claim that argument and interpretation are similarly related. My main reasons for this claim are as follows: (a) in the domain of archaeology there is rarely disagreement about facts but about interpretations of facts; and, (b) since interpretation is an open-ended process - any interpretation can be superseded - then the attempt to get to the correct interpretation requires argument and debate. The former is an empirical claim about what archaeologists do (though surely generalizable to many other sciences and to areas such as medicine, psychotherapy and so on). The latter is a claim about the nature of interpretation. If these assumptions are correct then argument needs interpretation to provide its object or topics whereas interpretation needs argument as the means by which more and more satisfactory interpretations are achieved. Apart from this an interpretation may be represented in terms of a claim with its supporting grounds and can therefore act as an argument-1.

In the remainder of this chapter I will present a model of interpretation and of argumentation; introduce the notion of argument style; discuss arguments from analogy and principles; and end with some comments on relevance and level shift in arguments.

4.1 Interpretation

A model of interpretation is needed as part of the overall model of argumentation since we need to specify the component which produces the object for argumentation. In the

domain of archaeology the principal reasoning task is interpretation and the principal object of argumentation the results of interpretation. Apart from excavation, there are others such as cataloguing, description and statistical analysis. In general, however, these either contain interpretative elements or can be seen as part of the interpretative process. While the model of interpretation forms part of my overall model of argumentation it can also stand on its own as a model of the interpretative task. For this reason I have discussed it in a separate section.

I have derived the model of interpretative reasoning used from the work of J-C Gardin (Gardin 1980, 1987) and various of his colleagues in which he puts forward what he calls a *logicist analysis* of archaeological reasoning. We have already discussed this work and its implications for archaeological reasoning in chapter 2. Here we will discuss the model of interpretation put forward.

Gardin's logicist models provide a clear, principled and general statement of interpretative reasoning. They are derived from a consideration of archaeological reasoning and therefore are ideal for the model of interpretation needed for a model of argumentation in that domain. At the same time they are of wider application. The approach has been applied to many areas both inside archaeology and outside. For instance there is an interesting analysis of Baudelaire's poem *Chats* in Natali-Smit (1981).

The logicist approach produces analyses of archaeological reasoning which can be expressed clearly in diagrammatic form. The principal material is the reasoning of archaeologists as exhibited in their texts. Thus the logicist model does not attempt to simulate the actual process of forming the interpretation but the process of justifying it. Logicist models 'are not meant to provide a film of the successive stages of a construction, as they 'really' developed, but rather a kind of flow diagram indicating one way of reproducing the construction through a mechanistic sequence of operations that may have nothing to do with the author's own mental processes' (Gardin 1980: 122). They are thus necessarily related to justification and hence to argument.

The main aim is to produce an architecture of the propositions which make up an interpretation. There are, as we have seen, three types of proposition: (a) those which capture the initial data forming the basis for an interpretation or Po propositions; (b) those which capture the end product of the process of reasoning involved in the interpretation (Pn); and, (c) the intermediate propositions which capture the stages in the reasoning intermediate between initial and terminal propositions (Pi). A tree-like

representation of an interpretation can be produced in which the connections between nodes are determined by the application of transformation rules. The type of the resulting relation between the nodes depends on the type of the transformation. Possible transformations include various forms of reformulation and of inference. For instance a transformation may be applied to a node to derive a deductively valid proposition which can serve as a node at a higher level. Here the relation is one of implication. Or again a transformation may be applied which produces semantically equivalent values. The relation being one of sameness.

We have already given an example of the use of the approach in Figure 2.1. In this example the Po, Pi and Pn propositions are clearly distinguished. We can also see the different levels of interpretation as the interpretation becomes more and more precise. The sorts of transformations which are operative can be expressed as rules in a production system. Gardin et al. (1987) and Lagrange and Renaud (1985) describe the results of experiments in the use of production rules to capture the transformations. These rules can be exemplified by the following:

If
 X is holding a hawk in a scene
Then
 The scene is a hunting scene.

In our application of this model, archaeological interpretation is taken to have the following components (see Figure 4.1):

- 1a. *Classification* of features. The term "features" encompasses the various aspects of archaeological sites both man-made and natural (e.g., pits, ditches, walls etc.).
- 1b. *Classification* of finds. "Finds" are mobile features which are either man-made or natural such as bone fragments.
2. The *reconstruction* of past human activities in terms of *activity areas* and their associated activity. An "activity area" is a significant area of a site at which identifiable human activities (e.g., cooking or hide-working) were carried out.
3. *Cultural interpretation*. That is, the creation of an interpretation or cultural profile for the site as a whole which includes a determination of the technology, subsistence, social organization and religious or other beliefs of the occupants of the site.

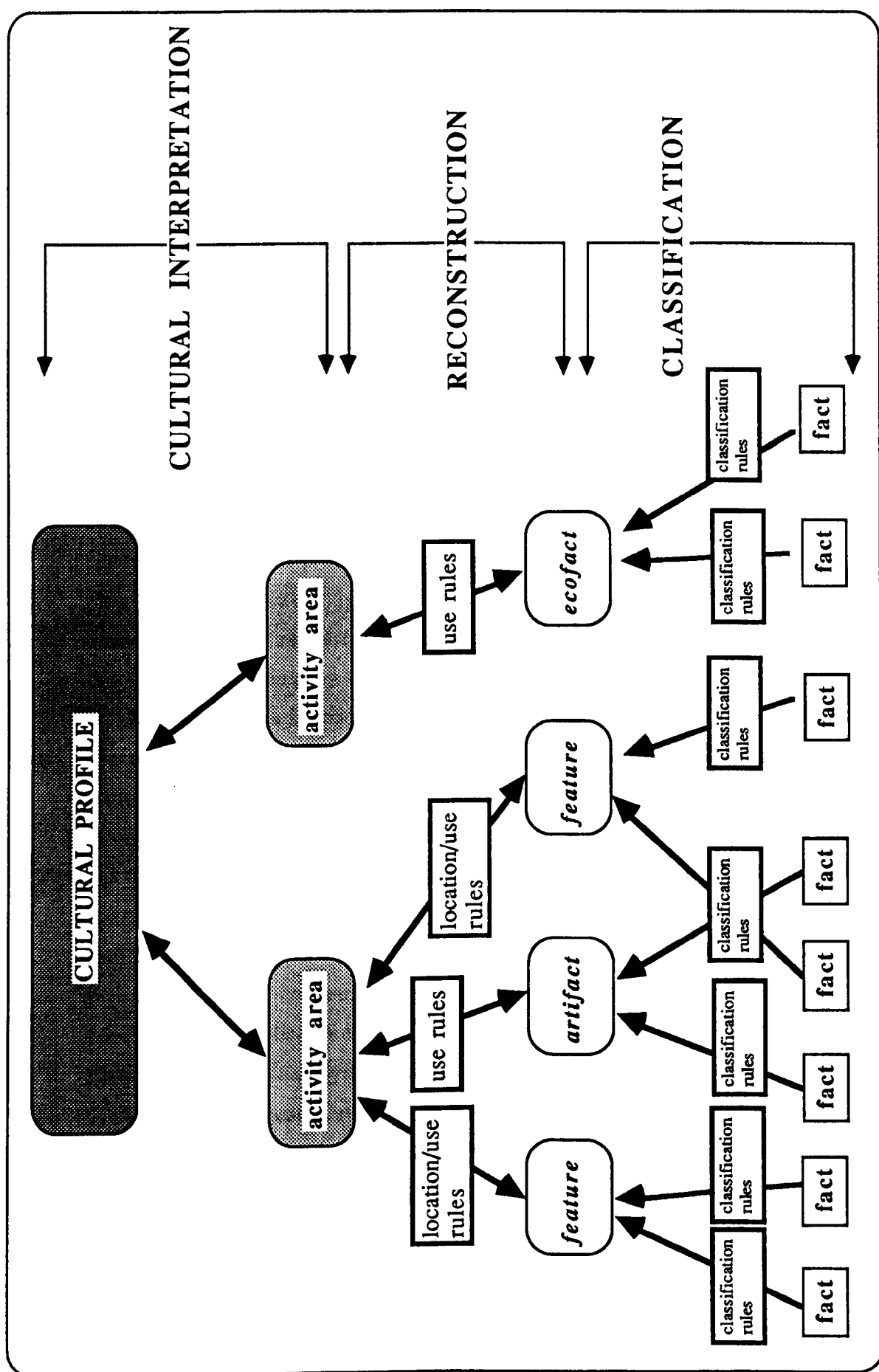


Figure 4.1 - Archaeological interpretation

Thus the application of the transforms or rules produces higher and higher levels of interpretation. At each level a clear account can be given of all the propositions involved as well as all the logical and semantic links. We can illustrate the process of interpretation involved here by an example. Suppose that in a given site we have found the following:

- a ring of stones
- a pit within the ring of stones
- a spatially related stone artifact with sharp elongated edge

During the process of *classification* we can use these finds to infer, for example, that there is a firepit on the site (a pit surrounded by a ring of stones) with a knife (a stone artifact with a sharp elongated edge) related to it. Using this information and our knowledge that knives may be used for cooking or hide-working we can provide a *reconstruction* (or *Pi* propositions to use Gardin's term) of the area around the fire-pit as either a hide-working area or a cooking area. This, in turn, can provide the basis for a *cultural reconstruction* at a higher level, for example, that the economy of the site is predominantly hunter-gatherer.

As it stands this is too simple a model of interpretation. The above example illustrates how the model can be used to derive conclusions or *Pn* propositions in a bottom-up or data-driven manner. However, archaeological interpretation is more likely to proceed in a top down manner. When an archaeologist interprets a site s/he has a model of the kind of site which determines what s/he *expects* to find. Thus in the example given above, the archaeologist, thinking that s/he is excavating a Pueblo Indian site, will expect a particular range of artifacts and features. Gardin recognises that his model can accommodate top-down or hypothesis-driven reasoning but considers these to be merely alternative ways of looking at the interpretation. However, if logicist analysis is to function as a model of the process of interpretation as opposed to the justification of interpretation it must incorporate a top down element since the production of interpretations in archaeology will involve abductions or guesses as a means of explaining the occurrence of particular anomalous results. As Schank (1986a) suggests, we only need explanations when our expectations are thwarted in some way. Otherwise we can categorise data readily. In the same way we really only need to provide interpretations when an anomaly occurs. For instance,

when an unexpected artefact is found. Winograd makes a similar point when he suggests that reasoning is triggered when some new piece of information has led us to question something previously taken for granted (Winograd 1980: 235). If we incorporate this top down element we produce a model of interpretation in which propositions are produced initially either by applying rules in a bottom up manner or by making use of an ideal description of a site. Each node may then be subject to several cycles of bottom up reasoning and top-down checking before the final value is arrived at. This may involve a reappraisal of all the lower nodes in the tree which constitutes the partial interpretation. Thus, by gradually extending the satisfactory nodes the final interpretation is produced.

This expectation-based model of interpretation has been implemented for the domain of archaeology and reported in Patel and Stutt (1988, 1989). The system we produced was called KIVA and reasoned about the domain of Pueblo Indians from the south-west of the USA. (A kiva is a ceremonial area in a pueblo village.) While this approach has close affinities to the model-based approach advocated by Clancey (1986b) as well as to the techniques for combining different kinds of knowledge in the domain of language understanding (e.g. in Hearsay-II, Erman, Hayes-Roth, Lesser and Reddy 1980) we chose the term expectation-based rather than model-based since in the actual implementation there is only an implicit model of an archaeological site. This is distributed in a set of rules which act as constraints on the possible interpretations. Typically a series of interpretations would be produced which could be pruned subsequently by the application of the constraint rules. KIVA builds up all possible solutions and, from its knowledge of a typical site, picks out the best solution (or solutions). In the above example the system could apply a set of constraint rules which includes the knowledge that all Pueblo sites have a cooking area. Thus it could determine that it is better to believe that the activity carried out in this particular area was cooking since the area has a firepit and no other area of the site has evidence for this necessary component of a site of this kind. Furthermore, from its knowledge that hide-working was never carried out in an area reserved for cooking, it could determine that cooking was the *only* activity carried out. (I shall have more to say about these rules in the next chapter.)

While the expectation-based approach is intended to model the reasoning of archaeologists in discovering an interpretation, it can also serve as a means of dealing with uncertainty in archaeological reasoning. Uncertainty in archaeological reasoning

arises in at least two ways:

- (a) That which arises because more than one interpretation of the data will always be possible. As in everyday reasoning, the inferences that are made are usually plausible rather than certain. This is because there is inevitably a gap between the material evidence and the interpretations placed upon it. While there have been attempts to produce law-like generalizations in archaeology (e.g., Schiffer 1976), these have mostly been confined to the lower levels of interpretation and there is some doubt about their applicability even at this level.
- b) That which arises because the data is incomplete. Since there is a limited supply of data available and this may be destroyed in the process of excavation which precedes interpretation, the evidence upon which the archaeologist bases his or her interpretation will always be radically incomplete. This incompleteness in the data will result in uncertain inferences.

Unlike the alternative approaches to uncertainty, the expectation-based approach deals with both the multiplicity of contending interpretations and the incompleteness of the data. In the case of the former, the site model can be used to reduce the number of interpretations. In the case of the latter, the model can be used to make plausible assumptions which will allow reasoning to proceed even when data is missing. In our implementation, we concentrated on the former. Moreover, since the knowledge which is used to select the possible interpretations is represented explicitly (in what is, in effect, a distributed model of a typical archaeological site) the knowledge of how the system reached its decision about its reasoning is available for possible use in explanation.

There are still inadequacies in the above model. For instance there is no means of representing the different theoretical backing which may govern the application of a transformation and hence no way of capturing the notion of a theory determined interpretation. There is also no means of distinguishing between types of inference. As we shall see below, the Toulmin method of analysing arguments gives us a clear means of representing transformation rules and their backing. In section 4.3 the use of this in arguing will be explored. In effect, my work can be seen as an attempt to integrate the work of Toulmin and Gardin within the framework provided by a model of argumentation.

Another feature missing from the above account is explored by Winograd and

Flores (Winograd 1980, Winograd and Flores 1986). This is the notion of *pre-understanding* which derives from the work of philosophers such as Heidegger and Wittgenstein.

In any situation where we are interpreting language, we begin with a system of understanding that provides a basis within which to generate an interpretation. This pre-understanding in turn arises and evolves through acts of interpretation. This circle, in which understanding is necessary for interpretation, which in turn creates understanding, is called the *hermeneutic circle*. (Winograd 1980:223)

Winograd (1980) is happy to equate this with the computational use of frames which are partially filled to begin with and filled more completely as processing proceeds. Winograd and Flores (1986) use the need for pre-understanding (or background) as important plank in their argument against the possibility of a computational language understander which we will discuss in the final chapter. The important point to note is that all interpretation requires knowledge. This insight is similar to that of researchers in explanation based learning (for example, Mitchell, Keller and Kedar-Cabelli, 1986) where prior theoretical knowledge is used in learning new concepts. In fact this too can be construed as a form of interpretation. The other point to make is that all interpretation takes place within a background or context. This context is partially given by the facts at issue. In our case, they are given by the domain (and sub-domains) of archaeology. The context is also partially determined by the knowledge which is brought to bear on the facts and (as we shall see) the kinds of inferences made and the backing for these inferences.

Taken simplistically the outlined model can be seen as another way of stating the nature of any expert system as the application of rules to initial data to produce structured sets of propositions. Of course the inference engine in an expert system is sometimes called an interpreter but this is to be distinguished from the high level task of interpretation which, according to Stefik et al. (1982), is one of the main expert tasks which an expert system might be expected to perform. They define it (ibid: 136) as 'the analysis of data to determine their meaning'. The work of Lagrange and Renaud and others suggests that the approach is readily applicable to the construction of expert systems. However while all expert systems may be described as producing transformations of data they are not aimed at producing the structured output of a system based on the above principles. The logicist approach is very similar to the discussion of BLAH (Chapter 3 section 3.1.2) where the system produces explanations as the result of transformations on a tree. Both are equally compelling

and similarly tractable models. In fact, the extended logicist model provides a new model for the interpretation task in expert systems. This especially true if some mechanism for explicit and justifiable choices between the possible interpretations is incorporated. With this the internal reasoning of the system more closely matches its external role as a provider of interpretations, explanations (or arguments) as well as decisions. The internal process involves the production of interpretations followed by the justifiable choice of one of these. In this case the system can be said to argue with itself and explaining or justifying becomes the dominant reasoning task rather than something which is tacked onto the reasoning. We have already met this suggestion in our discussion of BLAH where Goguen et al. (1983) define reasoning as the internal form of that which is expressed outwardly in language as explanations. Other researchers have made similar claims. For instance Winograd: "...reasoning is a form of arguing with yourself, and the justification for a step is that it (temporarily at least) quiets objections to some statement" (1980:235). There is also work on the explicit representation of justifications for knowledge base refinement (Smith, Mitchell, Winston and Buchanan 1985). Thus explanation, justification and argument may not be seen as something which is additional to reasoning. They may be seen as one (or the) form of reasoning. This is especially true if one eschews the logical approach to belief change as, for instance, Harman (1986) does. Here reasoning has no relation to logic but is rather thought of as principled change of view.

In conclusion of this discussion, logicist models succeed as a means of analysing the reasoning of archaeologists as well as much of our everyday reasoning. This is shown by the success of systems built in terms of the logicist approach. These systems reveal aspects of the process of interpretation. On the other hand the logicist model needs to be augmented so that it can deal with discovery as well as justification. By adding a top-down component, the model not only becomes psychologically more plausible it also provides a means of dealing with uncertainty. The basic logicist model also needs to be augmented so that the different types of reasoning can be clearly distinguished and so that backing theories can be dealt with. Finally the range of choices available at any node in the interpretation tree necessitates the addition to the model of a mechanism for dealing with these. While the expectation-based approach can prune these, a computational arguer provides a mechanism for comparing the various interpretations, deciding between them, arguing for or against them and storing the resulting argument-2. Thus the inevitability of

choice coupled with the need for explanation or justification requires that the model includes argumentation. Thus, interpretation, properly viewed, intersects with argument. In the next section we shall see how this model of interpretation provides the means for the production of argument-0s and hence argument-1s.

4.2 Argumentation

*One school thinks a post-hole in an ancient floor
stands first of all for a pupil in an iris.
The other thinks a post-hole is a post-hole. And so on -*

*like the subversives and collaborators
always vying with a fierce possessiveness
for the right to set 'the island story' straight.* Seamus Heaney - Parable Island

In this section I discuss the model of argumentation which I have employed. I first of all discuss the work of Stephen Toulmin and then put forward my own model of arguing based on this work.

4.2.1 The work of Toulmin

Toulmin's 1958 book on the uses of argument has long been one of the most influential texts in this field. This influence is apparent not only among informal logicians or theorists of argumentation but also among researchers interested in the computational modelling of argument and, more recently, among explanation and dialogue theorists.

In outline, Toulmin's discussion of the correct model for arguing moves from a rejection of logic as a model to the use of legal procedures.

There has been much debate over the years over the precise nature of human reasoning (Henle, 1962, Johnson-Laird, 1983, 1987, Rips, 1983, Cheng and Holyoak, 1985, Cheng, Holyoak, Nisbett and Oliver, 1986, Lakoff and Johnson 1980a, 1980b, Lakoff 1987). It has been argued on the one hand that human reasoning is conducted much in the way that a logical proof is constructed so that formal logic, while representing an ideal, is nonetheless a good model of human reasoning. On the other hand it has been suggested that formal logic is a normative rather than a descriptive discipline and that it relates only to reasoning of a very specialized and abstract kind. I am of the latter opinion. One reason for believing this is the bizarrerie of material

implication, which has the following truth table:

A	B	A -> B
=====		
T	T	T
T	F	F
F	T	T
F	F	T

This table expresses the notion that the implication holds except in cases where A is true and B is false. As a result of this 'a false proposition materially implies any proposition...and any proposition materially implies a true proposition' (Lacey, 1976). There is nothing in the definition captured in this table which requires that the propositions have any relevance to each other. Thus the proposition "The moon is made of green cheese" materially implies the proposition "I am constructing an example for my thesis".

Another is the empirical evidence for the difficulties that novices have in getting to grips with the logic programming language Prolog (Taylor, 1984). Winograd also agrees: 'In looking at any significant sample of natural language, it becomes quickly apparent that only a small fraction of human "reasoning" fits the mold of deductive logic' (1980: 219).

Logic and argument are two separate though interrelated areas. Argument is the process by which two or more agents arrive at the truth of something by an adversarial process. Logic is a normative system related to mathematics for the manipulation of propositions which have a particular form. Of course we sometimes use deduction in argument but an argument can proceed without logic in its formal sense. Auguste Dupin (Poe's detective) or Sherlock Holmes may claim that it is logic which helps them solve various mysteries such as the murders in the Rue Morgue or the Silver Blaze mystery. However, while they make use of logical elements, they principally employ a process which Pierce calls *abduction* - i.e. best-guessing or hypothesis making given the evidence. They interpret the evidence and find an *explanation* which explains all or most of the facts presented. This process cannot be represented in terms of logical proofs. Of course story-tellers cheat and coincidences and fortuitous happenings will occur which allow the detective to present his

interpretation of the case. For instance the broken nail in the murders in the Rue Morgue. So detectives are not perhaps the best analogues to arguers. For more on the relation of fictional detectives to abduction see Eco and Sebeok on Holmes (1983).

If we no longer look to formal logic for our model of human reasoning where should we search? Toulmin encountered similar difficulties in his analysis of arguments. As he says (1958:2):

In fact ... the science of logic has throughout its history tended to develop in a direction leading away ... from practical questions about the manner in which we have occasion to handle and criticise arguments in different fields, and towards a condition of complete autonomy, in which logic becomes a theoretical study of its own, as free from immediate practical concerns as is some branch of pure mathematics ...

His solution to the problem of analysing and assessing arguments was to turn to jurisprudence or the discussion of the philosophical and theoretical issues behind the law. Toulmin uses the 'jurisprudential analogy' as a means of bringing out the notion that what we are concerned with is reasoning as a critical faculty.

A sound argument, a well-grounded or firmly-backed claim, is one which will stand up to criticism, one for which a case can be presented coming up to the standard required if it is to deserve a favourable verdict (1958:8).

Thus the standards which logic gives us may play some role in everyday argument. However Toulmin uses his analogy to bring out another point. This is that in different types of legal cases what counts as evidence and how this is related to the claims made will vary. Different evidence may be needed for a civil case than for a criminal case. Toulmin extends this to everyday arguments. He calls the features which do not vary with different fields of argument the 'field-invariant' features. The former include the force of an argument, its formal structure and the stages through which it develops (see Rowland, 1982). By the force of a term Toulmin means 'the practical implications of its use'. Toulmin uses the term 'cannot' in illustrating this notion. The force of the term is the same in any domain. However Toulmin would argue that the criteria and standards for its use vary across different domains. The criteria for the correct application of 'cannot' in a game are different from those in common sense reasoning. This can be exemplified by the contrast between 'If it is X's move, Y *cannot* move' and 'The sun *cannot* go from west to east during the day'. These criteria for assessment along with different data and the different types of claim-data relations are what Toulmin calls 'field-dependent'. Thus from the legal model Toulmin derives a notion of argument in which procedure is as important as standards and in

which different domains have different ways of arguing and hence different modes of assessment. Toulmin's model of argument thus goes beyond narrow logical models of argument. Whatever the merits of logic for representation it has no greater claim for the formalizing of thought than have other models such as the present one of legal argumentation.

In line with this Toulmin produces a more complex analysis of the components of an argument. In what follows I shall refer to this as a *toulmin structure* and to the graphical representation of it as a *graphical toulmin structure*. This analysis is in terms of six components. These can be illustrated as in Figure 4.2. The principal components are:

- The *datum* or evidence which we use in our argument;
- The *claim* which we make as the outcome of our argument;
- The *warrant* which allows us to derive the claim from the datum;
- The *backing* or grounding for the warrant;
- The *model qualifier* which shows how strong the derivation of claim from datum is;
- The *conditions of rebuttal* which indicate when the derivation does not apply.

This can be illustrated with an example drawn from Toulmin (1958: 105)

Claim:	Harry is a British subject.
Qualifier:	Presumably.
Datum:	Harry was born in Bermuda.
Warrant:	A man born in Bermuda will generally be a British subject.
Backing:	The relevant statutes and other legal provisions.
Rebuttal:	Unless both his parents were aliens or he has become a naturalised American

Another, more recent example, comes from Toulmin, Rieke and Janik (1979: 253).

Here they present as a typical scientific argument the following:

Claim:	The earliest known anthropoid apes lived in the African Rift Valley.
Qualifier:	Evidently.
Datum:	Geological and palaeontological reports from Africa, China, Java, etc.
Backing:	Our experience in developing a systematic interpretation of palaeontological evidence indicates that
Warrant:	The presence of fossilized anthropoid remains in earlier rock

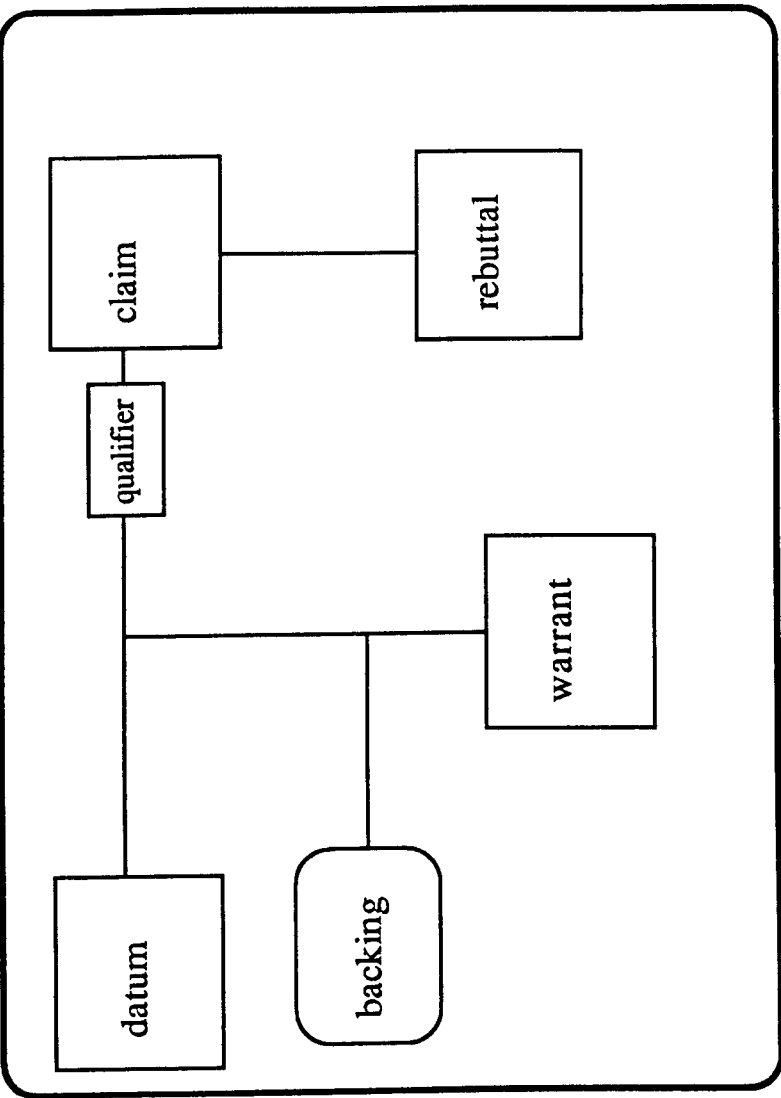


Figure 4.2 - A full Toulmin structure

formations in one place than in another indicates the earlier existence of living anthropoids in the one place than in the other.

There is no *rebuttal* given for this since, as they say (ibid: 255):

[A]rguments between paleontologists about the chronology and genealogy of the early anthropoid apes often turn on the respective plausibilities of rival - and sometimes contrary - interpretations. Given the available observations, any claim in this field can afford to be presented with some modesty and appropriate modal qualifiers... Correspondingly the unavoidable inconclusiveness of arguments in this field shows up also in the variety of possible rebuttals that might be advanced against any particular interpretation.

These remarks could equally well be applied to archaeologists.

While this analysis predates the use of knowledge based systems by some decades it is easy to see how the above could be mapped onto the components of an expert system. The datum and claim represent the antecedent and consequent of a production rule which itself is the warrant. The modal qualifier is the certainty factor. More recent work on the design of expert systems even makes use of explicit representations of conditions under which the rule will fail (e.g. Mott and Brooke, 1987). The only element which does not have a role in expert systems is the backing. As we shall see below this forms an important element in my own work.

In this work I have made no use of the modal qualifiers or conditions of rebuttal. In the case of the former this is because the opportunity for ongoing argumentation circumvents the need for the calculation and presentation of qualifiers; all arguments are considered to be provisional. I have avoided rebuttals since, as the above quote suggests, in many domains there will be too many possible ways of rebutting an argument. Furthermore I prefer to use the term *ground* rather than *datum* for the evidence. Thus Figure 4.3 represents the simplified structure I shall use. This figure also illustrates some of the different types of warrant which can be used as well as the different kinds of backing. These will be dealt with in section 4.3. They are introduced here to show the value of toulmin structures as an easy way of encapsulating the various elements. As we shall see, when represented computationally as frames they provide a simple means of moving between the various elements.

Although Toulmin stresses the procedural notion of legal argumentation he is more concerned with the structure and assessment of arguments than with these procedures. He is not at all concerned with the actors in the various procedures nor with the nature of the dialogues into which they enter. In attempting a computational

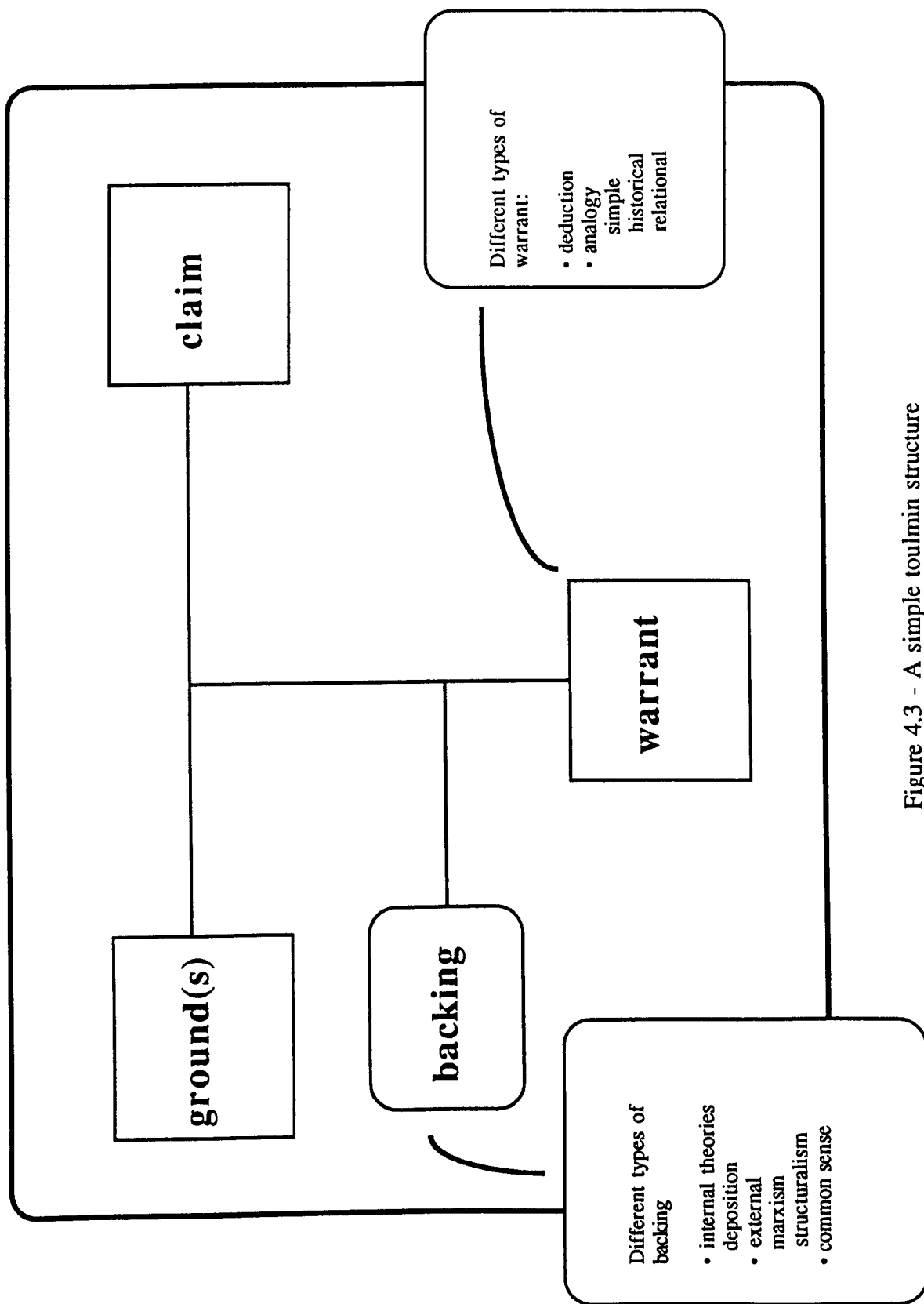


Figure 4.3 - A simple toulmin structure

model of argumentation it is perhaps necessary to take account of the participants as well as the arguments they produce. These components will be dealt with in my own model below.

4.2.2 A simple model of argumentation

If an exchange between two participants is to be regarded as an argument it must have at least the following components. Human argument exchanges will include other components some of which, such as common sense knowledge, are not easily captured on a computer.

- Argument is an *exchange* between two or more participants. As such it must have a *symbolic medium* capable of sustaining at least an approximation to natural language interchanges.
- An argument, like other conversations, is composed of *moves* (cf Reichman-Adar 1984). A move is a description of a chunk of the interchange which is described in terms of the purpose of the agent instigating the move. For instance, during the course of an argument, a participant will attempt to defend a claim s/he has made by means of a *support* move.
- These moves have *rules* which govern whether or not they are regarded as legal by the participants.
- Participants take *turns*. In any well ordered debate an implicit mechanism for turn taking will be in operation.
- During a move an *argument step* is produced. By argument step I mean the set of related propositions which make up the contents of a move in an argument exchange. An argument step is composed of a claim with supporting grounds. Grounds can be related to claims by warrants or derivation rules drawn from common sense or accepted as conventional in the particular domain. The warrant can be given a backing or statement of provenance or authority (Toulmin et al. 1979). Note that grounds, backing and warrants can in their turn become objects for argumentation. Thus we could have an argument in support of a particular backing. Grounds can of course be

the claims of sub-arguments and so on.

- An *argument exchange* or debate is composed of a sequence of these steps which may be spread over several turns in the argument. There are relations of *support* or *attack* between the claims of previous and subsequent steps.
- One of the participants can *win* the argument. This should result in a change in the beliefs of the loser. Since we are not ideal reasoners it is rarely the case that human/human arguments have this result.
- If a participant can win/lose, s/he can also have *strategies* or plans which represent a means to the goal of winning. Conversely the other participant can have counter-plans which attempt to subvert those of his or her antagonist.
- Again, if there are to be winners and losers, there must be a means of *assessing* the strength of the overall argument.
- Arguments can be said to represent the *viewpoint* of the arguer. They are also frequently argued from some other point of view adopted for some specific occasion or in responding to a particular antagonist.
- It is possible to distinguish different *types of argument* depending on the type of reasoning involved. Thus we have deductive, causal and analogical arguments. Most arguments, however, are composed of mixtures of the different types of reasoning.
- Humans commit to *memory* at least the gist of arguments, standard patterns of argument for an academic discipline (such as archaeology) and good or successful arguments.

4.2.3 An improved model of argumentation

The above seems unintegrated and *ad hoc*. Perhaps it is not possible to construct a structure for argumentation which has the required rigour and includes all the seemingly diverse sense of the term from logical proof to a quarrel in a pub. In this section I will elaborate the above model so as to make the relations between the

various components somewhat clearer.

Following the work of O'Keefe (1977), Willard (1979a, 1979b) and Rowland (1987) on distinguishing between argument as a conversational interaction and argument as a set of propositions (and from which the convention of referring to these using numerical suffixes derives), I have formed a tri-partite model of argumentation (See Figure 4.4). This tripartite model of argument has much in common with the systems approach to the modelling of argument adopted by Pfau, Thomas and Ulrich (1987). Here the principal components are advocates, opponents, allies and critics. However, the components of my model are not simply the agents in the interaction but include the result of the interaction (the argument) and the processes which produce it. According to the theory there are three levels for discussing argument:

- Argument-2 where argument is viewed as a conversation or debate;
- Argument-1 where argument is viewed as a set of propositions in which one proposition (the claim) depends on the others (perhaps a series of sub-claims).
- Argument-0 which is a step in the set of propositions which make up an argument-1 - an argument-0 can be viewed as a toulmin structure as discussed above.

The relations between these are straight-forward. An argument-2 is composed of a series of interchanges which make up one or more argument-1s. An argument-1 is composed of one or more argument-0s. Thus argumentation can be viewed as a coherent activity in which logical proofs (extremely deductive forms of argument-0s) can be linked to quarrels in the pub (extremely loose forms of argument-2s). This is shown in Table 4.1.

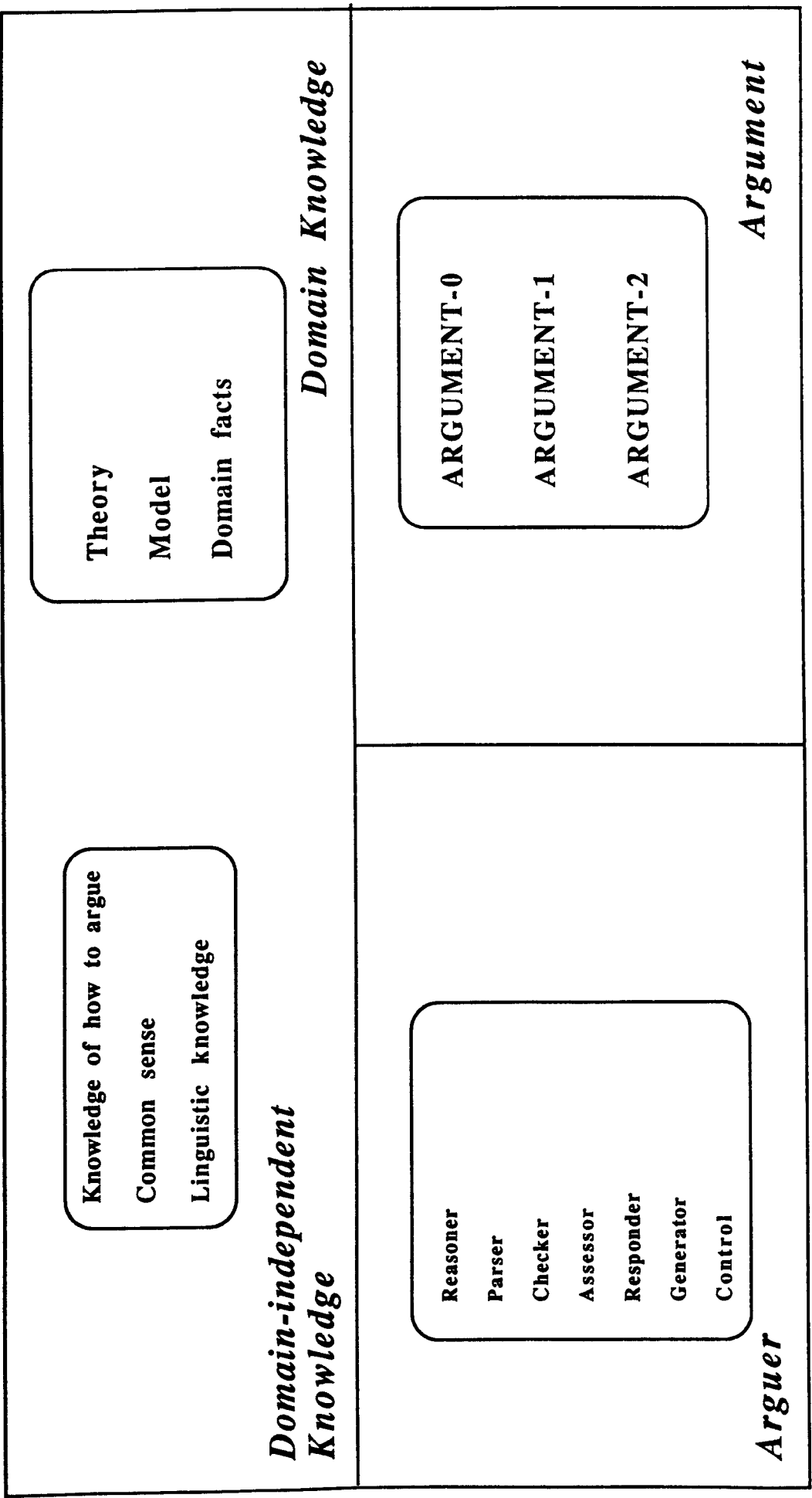


Figure 4.4 - The tri-partite model of argumentation

Argument level	formal	informal
=====		
Argument-0	step in proof mathematical calculation	step in interpretation section of conversation
Argument-1	proof	interpretation
Argument-2	mathematics debate	archaeological debate argument in pub

Table 4.1 - Formal and informal examples of the three levels of argument

This theory seems sufficiently general enough to capture the kinds of arguing done in archaeology where the argument-2s are well-disciplined debates but the argument-0s which are involved are largely non-deductive (see chapter 2 on archaeological reasoning). Indeed it is obviously general enough to capture many other domains (see chapter 7). At the same time we can see how the model of interpretation discussed above ties in to the model of argument deployed here. In fact this is quite straightforward: An interpretation viewed as above is simply an argument-1 composed of one or more argument-0s. Whether these are treated as argument or interpretation depends on the particular goals of the agent responsible. Nonetheless what is regarded as an interpretation insofar as it has some support can count also as a step in an on-going debate. In what follows I will elaborate upon this basic scheme.

As Figure 4.4 shows the model has three components: the arguers, the argument and the domain or field of argument. The argument component is itself threefold as we saw above and domains or argument fields can be construed as having three levels. In what follows I will elaborate upon this sketchy outline.

• Arguers

Each participant is composed of a reasoning mechanism. This reasoning mechanism embodies the model of interpretation discussed above along with extensions for dealing with analogy and arguments from principles to be discussed presently. Insofar as the model of interpretation only deals with a particular task, and there are different models for, say, explanation or planning, then the reasoner would have to embody

these as well. The reasoner also encompasses the various dynamic tasks involved in the production and response to arguments. These are parsing, checking, assessing, responding and generation. These tasks will be dealt with in more detail in the discussion of the program design in the next chapter. A control task is needed to schedule the various tasks. In pursuit of the various tasks the arguer needs to have access to various forms of knowledge. These include domain knowledge as well as knowledge of how to argue, common sense and linguistic knowledge.

- **Argument**

Arguments can be viewed at three levels as we have seen: argument-0, argument-1 and argument-2. An argument-0 represents a step in an argument-1. An argument-1 is equivalent to one of Gardin's interpretations. An argument-2 represents the on-going dialogue between two or more participants each of whom deploys a series of argument-1s. Each of the levels of argument has a characteristic graphical representation. We have already seen how these are related in Figure 1.1

- **Argument-0s, argument steps or Toulmin structures - Figures 3.2, 4.3**

These may be represented as in Toulmin, Rieke and Janik (1979) as a series of related propositions. An argument step is a structure which is composed of a claim and a set of propositions representing grounds, warrants or backing. Grounds can be related to claims by warrants or derivation rules drawn from common sense or accepted as conventional in the particular domain. The warrant can be given a backing or statement of provenance or authority.

- **Argument-1s or interpretations - Figure 4.5**

An interpretation is taken as a tree-like structure in which a top level claim (or claims) is supported by a series of propositions or grounds. Each of these sets {grounds, claim} constitutes an argument-0. Each of the grounds may in turn provide the claim for a sub-argument. The relations between the propositions are mediated by the warrants or rules and include relations based on deduction, analogy, causation and so on. An argument-1 may be composed of a single argument-0.

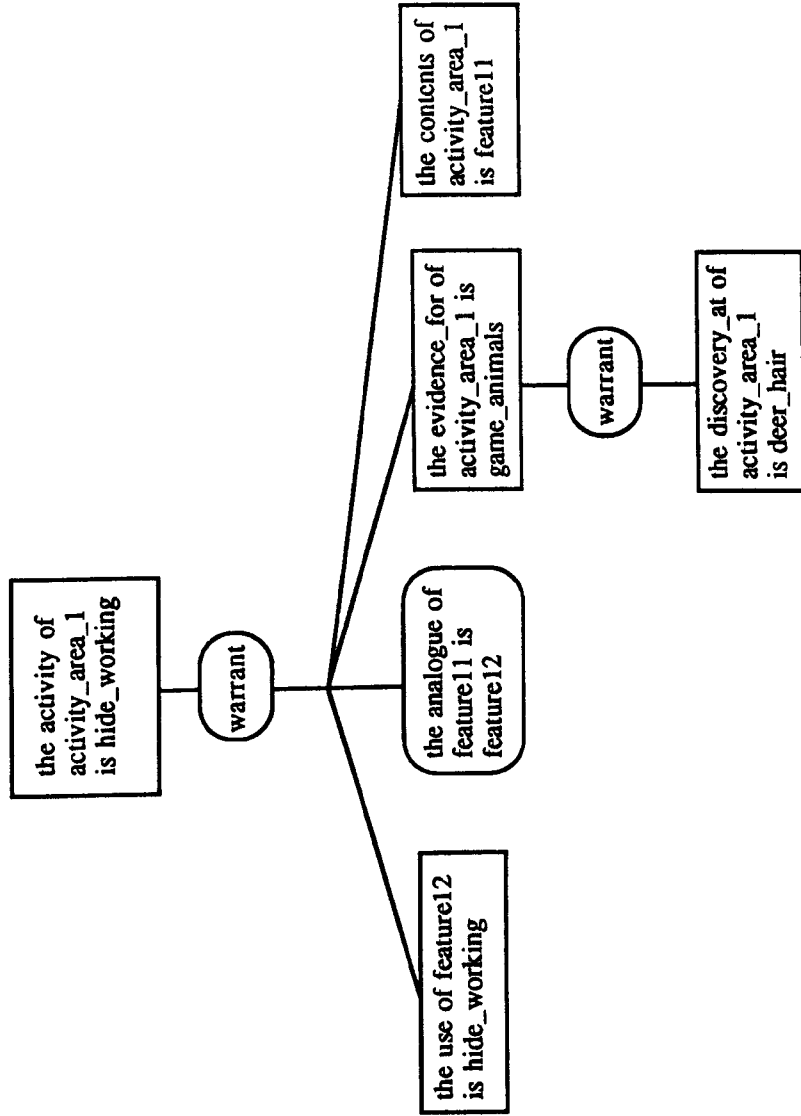


Figure 4.5 - An argument-1

- Argument-2s or debates - Figure 1.6

An argument exchange is taken to be composed of a sequence of argument-1s which may be spread over several turns in the argument. There are relations of support or attack between the claims of previous and subsequent steps (cf Flowers et al 1982). An argument-2 can be represented as a network.

The different operations possible at each level in the argument and the sorts of data structure they may be performed on are summarized in Table 4.2. This table will be expanded in the next chapter when we discuss a design based on this model.

level in model	structure	operations
=====		
argument-0	toulmin structure	reasoning/inference
argument-1	tree	interpretation
argument-2	network	argument/debate

Table 4.2 - Operations and data structures at each level

- domain

The model allows for argument between one or more participants about a particular domain. A domain or field can be thought of as having more than one level. As we have seen in discussing the Gardin work on interpretations a domain can be construed in terms of levels of interpretation. Thus there are likely to be rules appropriate for different levels of interpretation. The knowledge of a domain can also be categorised in terms of how structured the propositions are. Accordingly I have produced a *three level model of fields*.¹ According to this there are three levels at which a domain can be discussed: theory or principles; model; and, domain facts.

- By theory or principles I mean the set of theories or principles which lie behind all research efforts in a field. For instance in physics both the theories of relativity and quantum mechanics fulfill this role. In archaeology there are various high level theories such as marxism and structuralism. These theories constitute the backing for a toulmin structure or argument-0 in a particular field.

¹ I am indebted to my colleague Peter Cheng for long discussions on this topic. Many of the original notions were his though I have reworked them for my own ends.

- Intermediate between the notion of theory and domain knowledge there is the notion of a model. A model encapsulates some systematic relation between domain objects derived from the theory. It can be used to interpret or guide reasoning at the lower level. The model may be a propositional model or it may be a simulation or, as in the case of Crick and Watson's helical models of DNA, it may be physical object. In archaeology models, derived for instance from systems theory, are constructed to show how societal change may come about (for example, see the discussion of Doran's work in chapter 2 above).

- At the lowest level we have a set of data, relations between data and means for low level transformations of this data. For instance in physics observations and calculations will take place at this level. In archaeology this is the level of the basic domain facts and relations. These include information about the artifacts found on sites and the stratigraphical layers in which these are found.

Of course this three-level view of fields can be related back to our model of argument. The data forms the evidence or facts for an argument-0, while the lowest level operations provide the warrant for the argument-0s which make up the lowest level of interpretation or argument-1. A model can be expressed as a set of warrants or can provide the backing for warrants. Theories, as we have seen, provide the backing for warrants at all levels in the interpretations which make up the overall debate or argument-2.

The above view of argument is perhaps too weighted on the side of those who would hold that the propositional is the most fundamental; that is, that argument-2s are defined in terms of argument-1s. Others, especially Willard (1979a) have argued that the argument-1s depend on argument-2s. Or rather on the cognitive and affective states of the participants in an argument-2. '...arguments are unique interactions by virtue of what the arguers think they are doing... The best paradigm case of argument is a fleeting ephemeral experience which cannot be understood apart from the arguer's perspectives.' (1979a:25). While I accept some aspects of Willard's view I believe that there is a class of academic arguments or debates which can be characterized by the sets of propositions which make them up. Obviously this will

involve factors such as the affective states of the arguers and other tacit elements which are not captured in the propositional statements. However the most important elements of the propositions and their relationships will be captured. Another important element in Willard's discussion is the notion that arguments are context-dependent - 'The relevance of data and backing, then, can only be understood with reference to *specific encounters* which take their character from the constructions of the actors involved' (ibid:23). I will deal with the notion of context below. However, it must be pointed out that of necessity a computational means of dealing with context must miss out many of its elements; those which cannot be formalized (or, at least, not in advance).

As we shall see in chapter 7 the above model can deal with debates both within and without archaeology. For instance the continuing debate about the age of the earth can be seen as a series of argument-1s which go to make up an argument-2. The same is true of the debate about the origins of the wealth deposited in the barrows of the Wessex culture of the Bronze age in Britain which will be discussed in chapter 6.

A fuller model for argument-2s can be produced as in Figure 4.6. Here an argument-2 is concerned about some *topic*. The debate takes place within some argument field between two or more participants (or possibly, if viewed as a model of reasoning, between an agent and himself). The argument field is in turn situated within a shared culture. As a result of this many of the assumptions and common-sense notions are rendered tacit. An assessor (one of the participants or a formal referee) makes an assessment and decides on the winner.

Contexts are taken as the amalgam of the facts and theories which constitute the knowledge an agent has about a field or fields. Facts figure as data or grounds in the argument-0. Theories are accessed via the warrants. A warrant or rule can also form part of a model (e.g. of the growth of Wessex culture) and may provide inferences of various types such as analogy, causal, deductive and so on. Thus an argument-1, while largely drawing upon the context commonly used in the field (facts and theories) may make use of elements drawn from more than one sub-context. For instance an archaeological argument may depend on stratigraphical reasoning, on the results of a radiocarbon dating process, on analogies with modern non-industrialized peoples or on common-sense. Figure 4.7 shows this schematically.

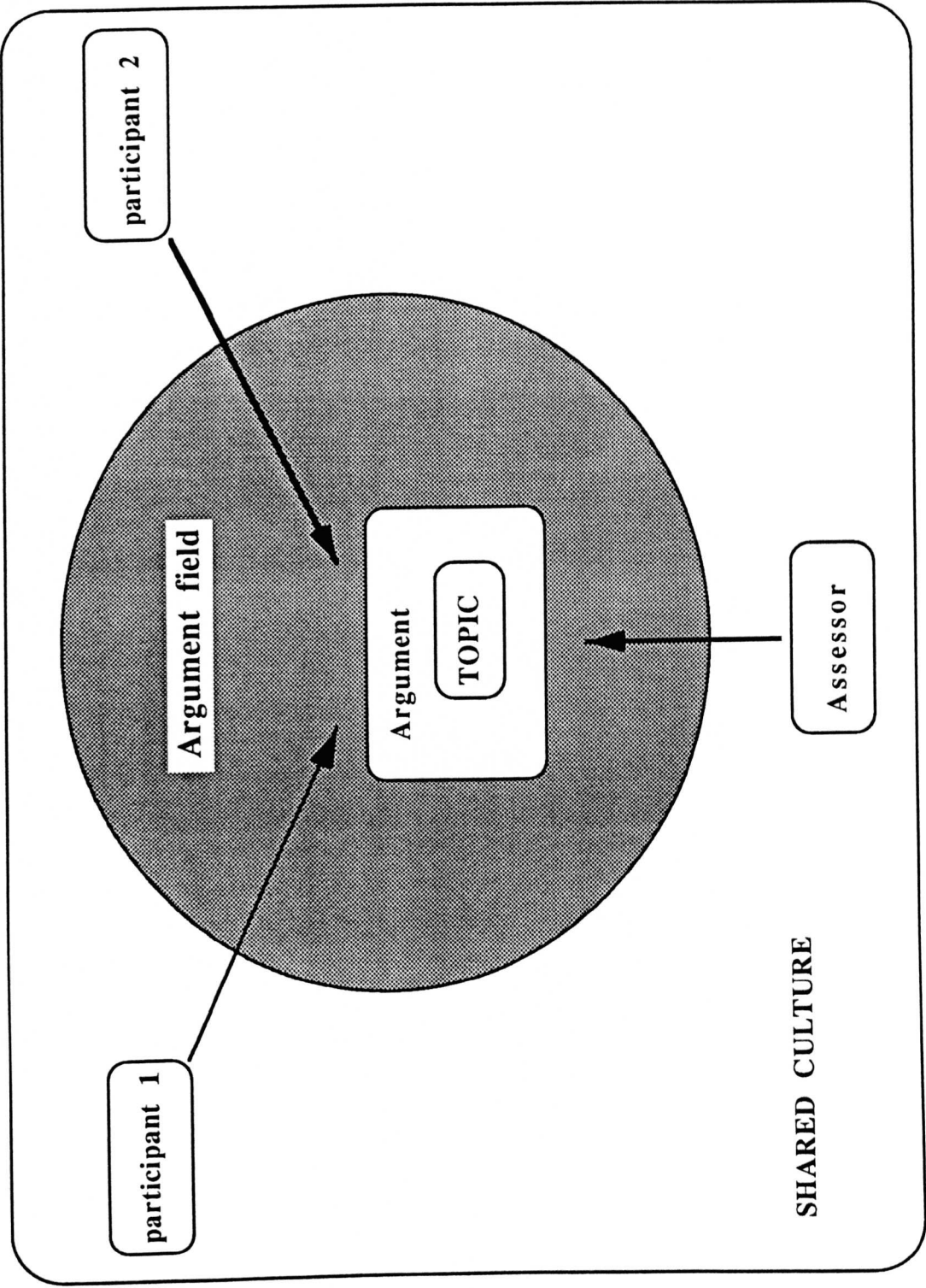


Figure 4.6 - Argument-2s and topics

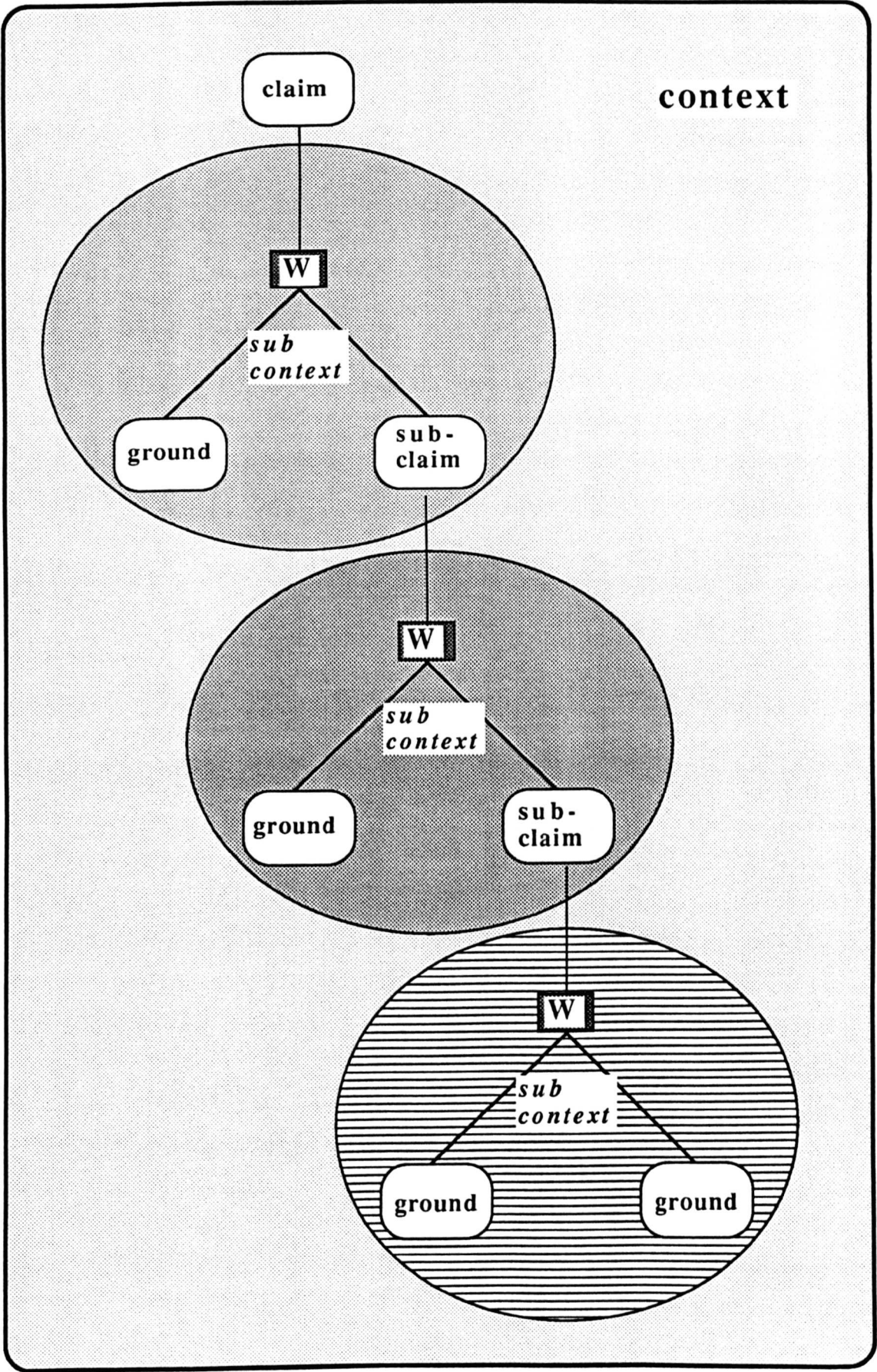


Figure 4.7 - Argument-1s and contexts

In order to deal adequately with argumentation it must be thought of as the action of a set of processes on a field. Thus when we discuss the implementation of the ASParch system below we will discuss both the domain and how it is organized and the argumentation processes which operate on this to produce an argument-2.

4.2.4 The uses of the model

The model outlined above has several important consequences. The most important of these are as follows:

- The model provides firm backing for the design to be presented in the next chapter as opposed to ad hoc construction.
- The model integrates interpretation and arguing.
- The model integrates different sorts of reasoning such as analogy and deduction in terms of different warrants.
- The model integrates theory and reasoning via the warrants and backing.
- The model clarifies the different operations and representations needed at each level.
- Relevance, context and topic shift are easier to deal with using this model as we shall see in section 4.4 below.
- Level shift can be dealt with by the extensions to the model discussed in the next section.

4.3 Argument style

While the above is a good general account of argumentation it is inadequate in many ways. As we shall see in chapter 8 there are many reasons why programs which attempt to simulate human processes such as the above are largely doomed to failure. The particular problem we are concerned with arises because, as we have seen in chapter 2, there are modes of reasoning which are specific to domains such as archaeology. The account given above shows what archaeological interpretation is and how it may relate to argument in archaeology. However, there is no systematic way in the above model of distinguishing the domain or field of archaeology from any other domain. The above model could apply to any domain. Of course this is a strength as well as a weakness as we shall see in chapter 7 below. However for the purposes of producing a program which can aid argumentation in archaeology it is necessary to find a way of specifying the particular kinds of argument common in

archaeology or, for that matter, any other domain. Thus, paradoxically, the addition of a means of specifying particular types of reasoning allows greater applicability of the model. Any system designed in terms of such a model should be able, as we are, to switch happily from arguments in one domain we are knowledgeable about to those in another. The knowledge of the domain is one aspect of this. And as Clancey and others have pointed out (see chapter 3 above) the different forms of knowledge in a domain must be explicit and separate. Here I am suggesting that yet another step must be made so that a domain or field has stored somewhere explicit knowledge of the types of argument which are possible. In order to deal with the differences between different domains I have introduced the notion of an argument *style*. An argument style identifies the types of warrant and fact which are commonly used in a domain. How relevant tokens of these are determined is discussed in the following section.

This section will discuss the following issues in the context of a notion of argument *style*:

- the identification of the different types of analogy appropriate to argumentation in different domains and the formation of appropriate responses
- the mechanisms for using models and theory to facilitate argumentation about a domain

Two main styles of argument will be introduced. Mode-based argument is primarily analysable in terms of its underlying reasoning style. Level-based argument depends on higher levels of model and theory. A variety of reasoning styles (such as analogy and deduction) are possible and hence a variety of mode-based arguments. There are only two level-based styles: those which depend on models and those which depend on more general theoretical principles. The discussion of argument style will be pursued with particular reference to the domain of archaeology and the design of ASParch. In the rest of this section I discuss what I mean by argument style, consider some of the mechanisms involved and mention some of the inadequacies they contain. In the following section I will discuss the notion of relevance as a counterpart to that of style.

4.3.1 What do I mean by argument style?

In linguistics, stylistics deals with the differing components which go to make up a particular author's style, or the style of language appropriate for different contexts.

Style is partially determined by subject matter. It also includes such things as the kinds of metaphor chosen, the tendency to use metaphor as opposed to metonymy, the particular vocabulary, the particular syntax and so on.

In argumentation a style involves a) the subject matter or the conceptual structure appropriate for a particular domain and b) different uses of different types of argument. Differences in the actual language used and in rhetorical devices may also play (a less significant) role. Canons of good reason (as opposed to canons of reason) are related to the style appropriate for a discipline.

The notion of an argument style derives from the seminal work of Toulmin (1958, Toulmin, Rieke and Janik 1979) discussed above and especially the notion of field-dependence. In his work on the uses of argument he argues that different disciplines have different canons of argument:

... all the canons for the criticism and assessment of arguments... are in practice field-dependent... we can ask, 'How strong a case can be made out?' ... and the question we ask will be how strong each case is when tested against its own appropriate standard. (Toulmin 1958:38)

We take Toulmin to mean that there are different kinds of argument for different disciplines (or fields) and that these need to be assessed differently. Since any discipline will in practice use a mixture of many of these kinds of argument, even if one dominates, we have adopted the notion of argument style. By this we mean the admixture of different argument kinds characteristic of a discipline. In archaeology the dominant argument kind is analogy. Given the lack of historical documents relating to remote periods in the past one of the principal means of acquiring knowledge about these periods is by analogy with modern non industrialized peoples. This use of analogical reasoning is a characteristic archaeology shares with the legal domain (MacCormick 1988, Adam and Taylor 1987).

At the same time a piece of archaeological argumentation will include other kinds of argument; for example, causal arguments about the natural processes which affect buried artefacts. Furthermore, another form of argumentation is frequently used by archaeologists i.e. arguments which are based on certain theoretical principles. There are for example marxist models of societal change as opposed to historical models in terms of invasions or some other mechanism.

Given the three-level model of fields discussed in section 4.2.3 above we can think

in terms of *inter* and *intra* level reasoning processes. Thus there can be a process of deduction using facts at the lowest level. We can deduce that since all flint tools will have traces of wear this tool will have traces of wear. We can also make deductions between levels. The level referred to need not necessarily be the next level up. An argument about the lowest level facts can make use of theories or principles. Thus in legal reasoning a case may depend on the facts plus an appeal to some high level principle such as 'freedom of speech'.

We can now draw up a table of a range of fields and their concomitant argument styles. In this table, the two main kinds of argument style - *mode based* and *level based* - are marked by the use of plain and italic text, respectively. A mode-based argument style is a simple admixture of argument types while a level-based style appeals to higher field levels.

Domain	Task	Style	Forum	Medium
=====				
Archaeology	Interpret	Analogy <i>Models</i> <i>Principles</i>	Papers	Language
Law	Trial	Analogy <i>Principles</i>	Court	Language
Lit. Crit.	Interpret	<i>Principles</i>	Papers	Language
Physics	Explain	Causal <i>Models</i>	Papers	Maths

Table 4.3 - A range of fields and their concomitant argument styles

Argument generation and comprehension as well as checking will depend on the dominant mode of argument for a field. If the dominant mode of an argument field is, as is the case with literary criticism, *principle-based reasoning* then there is little point in producing analogical arguments for users in that field. Nor will the system get far in understanding user arguments if it attempts to understand an analogy as a deduction and so on. In literary criticism long leaps to high level theories are the order of the day. In archaeology a system such as ASParch must be prepared to argue

analogically much of the time and to countenance appeals to models. This does not mean that there will *only* be analogical or model based arguments but that these are more likely.

As I have mentioned above the main point however in making this distinction (or series of distinctions) is that different argument styles have different methods of *assessment*. Thus a deductive argument should be amenable to a formal proof. An analogical argument should fulfill some of the criteria to be set out below and so on. In fact it will be the case that the comprehension of an argument depends on the checking of that argument. Thus a system will be aided in checking, comprehension and production if there is some classification of argument styles.

4.3.2 Computational mechanisms and responses

As an example of the notion of how argument style works in the domain of archaeology I will discuss the use of reasoning and extrinsic theories in what follows. A full computational model would have to provide means for all the different types of style possible. The following is meant to be representative. It is hoped that the discussion of these in this and later chapters will show how any component in a particular style might be implemented.

4.3.2.1 Analogies

Analogy has played a large part in the reasoning of archaeologists. However it is important to note that the primary use has been in determining the use or function of some artefact. The focus is mainly on the retrieval of analogies rather than their understanding (Gentner 1983) or the mapping of problem-solving knowledge (Holyoak et al. forthcoming, Keane 1985, 1988b, Kedar-Cabelli 1985). There are three principal sorts of analogy used in archaeology: simple, historical and relational.

A *simple analogy* represents the case where the attributes (or relations) of case C1 can be matched with those of case C2, thus allowing a future mapping of any attribute which may be of interest. In archaeology this will usually be the use or function of an artefact.

An *historical analogy* (of great importance to archaeology) represents the case where there is some historical continuity between C1 and C2. For instance it is possible to

make analogies between modern Hopi Indians and ancient Anasazi since it is believed that they are historically related. Keith Anderson (1969: 135) makes this point:

Important for interpretation of the Anasazi is the fact that they are the close ancestors of living Pueblo Indians. Anasazi cultural patterns are still preserved in the conservative Pueblo Indian villages of New Mexico and Arizona...For analogical interpretation of this well-preserved material, we can use the culture of the Hopis...

The third type of analogy, *relational*, owes its name to Ian Hodder's discussion of the theme in his book on the use of ethnography in archaeology (1982b). A relational² analogy is one in which there is some relationship between the attributes of each of the base and target pair. The most common form of relation will be derived from some account of how these attributes fit together into a nexus of material, social and psychological interactions. As Hodder says, '...I would search for a relational analogy which examined more adequately the relevant causal links between different parts of the analogy.' Interestingly, given the background of the different researchers, relational analogies have much in common with those dealt with by Gentner (1983) in her *structure-mapping theory of analogy* which has been mentioned above (in chapter 2). Gentner's favourite example is the analogy between the structure of the atom and the solar system. Here the success of the analogy depends on the second-order relation *cause* which relates first-order relations such as *attracts* and *revolves_around*. These interrelations in the solar system hold within the atom. Analogies of this kind have also been discussed by Wylie (1987) and Chouraqui (1985).

The rules for these analogies can be used as a means of generating arguments or in responding to arguments. Response to an argument is achieved by means of a strategy for response which deals with analogies used in argument. The most likely response is to examine the analogy for further attributes which lead to the rejection of the analogy. The following informal conversation shows one way of rejecting an analogy.

A: According to environmentalists the greenhouse effect means that the earth will

² The term 'relational' as Hodder uses it is not related to other more common uses within computer science - e.g., 'relational databases'.

rise several degrees in the next hundred years

F: If its a *greenhouse* effect won't that mean that it will get warm very quickly in summer and cold very quickly in winter

The response here attempts to show that if the analogy is taken seriously there is a contrary result to the usual accepted use of the analogy i.e. that there will be warming *and* cooling.

In general, there are two main means of responding to an analogy: (a) a rejection of the analogy because it is inadequate or because some feature of the base is ignored; (b) a rejection of the analogy because other analogies leading to contrary results can be found. These points have been made by Keane (1988a) when he says 'The central process in using analogies and counter-analogies is matching and establishing whether asserted correspondences are in fact valid and refuting them by using other analogies which highlight other aspects of the target domain'

The first kind of response to an analogy involves a process of checking the matching attributes and relations of X and Y. This check might show up a) wrong matches and b) significant features unmatched. Since nothing is ever a complete analogue of anything else (or it would be that thing) it will always be the case that only some of the attributes will match. Since this is so two conclusions follow: a) there are many potential analogues and b) important matches may not be made. Thus a response to an analogy in which Att1 of X is unmatched with anything in Y plus the claim that Att1 is significant is sufficient to count against the analogy and hence the argument as a whole.

The finding of an analogy to support a case involves the matching of attributes of the present case with those of a case which supports the claim required. In this process the finding/matching is *uncritical*. This can be modelled simply by finding the first simple analogue or in a more complex way by finding (or constructing) an analogy which satisfies some goal (Keane 1985, Kedar-Cabelli 1985). The responder to an argument must be more critical of the analogy. This involves an identification of the significant features of a case and an attempt to see if these are matchable. The related notion of relevance is discussed below. It is implausible to suggest that knowledge of what are significant attributes of a potential analogue can simply be stored by the implementer of a system. The system must be open-ended enough to cope with circumstances in which significance depends not on some overarching principle but on the particular point of the current argument. Thus while the date and

size of artefacts are always important in archaeology, in particular cases and for particular purposes, such attributes as provenance and material composition will become important. We suggest that significance is probably determined both by the goal (for example, refute the argument) and by the model or theory currently being used.

The second main way of responding to an analogy is by finding other analogies which have different implications. The following dialogue (freely adapted from Thouless 1930) illustrates this. Thouless calls this use of analogy *argument by forced analogy*.

A: We should cut welfare provisions. Look at lions they are magnificent animals because they are subject to natural selection.

B: But there are other species which have been produced by natural selection which are hardly magnificent. Sharks for instance or slime molds. Anyway, there are animals which exhibit altruistic behaviour towards the apparently weak and ill.

In this case the analogy (man::lion) is attacked not by finding deficiencies in the attribute matching but by finding an alternative analogy (man::altruistic animals) which tends to a diametrically opposed conclusion.

The notion of a sub-division of analogical arguments allows us to posit responses to the different types. This is something which has not been dealt with in other discussions of analogy in argumentation (e.g. Reichman's analysis (1981) of argument claim and challenge). For instance, simple analogies might be rejected for being simple analogies and therefore implausible. Historical analogies may be rejected for the lack of or weakness of the historical link. Relational analogies may be rejected for lacking some aspect of the causal linkages between attributes. Since there are no general laws in archaeology, it will always be possible to reject an analogy on this basis. This suggests that (as Hodder thinks) analogies in this third sense may need other forms of argument for their support.

4.3.2.2 Models/principles

Model or principle based argument has much in common with model based reasoning as discussed by Clancey (1986). Intuitively it is an attempt to capture the awareness that a) we use models/theories in our interpretation of the world and b) we rely on such theories when generating, comprehending and defending claims. (Similar work

on the simulation of political ideologies has been conducted by Abelson, 1973 and Carbonell, 1981. Ennals, 1986, has made suggestions as to how a computational politics might work). As part of the bottom up interpretation of an archaeological site various transformations or rules which are theory dependent may be used. That is to say that these have as their backing a body of theory of some kind. The set of rules can be construed as a model. In a top down mode, a theory may be used to generate the hypothesis which is tested by searching for the appropriate data. Such a hypothesis may also be generated in a bottom-up manner. If we have a more complex model (like that suggested in section 4.1 above) in which each node acts as a hypothesis created by reasoning bottom-up and verified or changed as the result of top down reasoning (which may lead to more bottom up reasoning and so on) theory may be said to play a role at almost every node in the network. Thus theory plays a major role in interpretation and hence in argumentation. As Shennan argues (1988):

A piece of work and the conclusions drawn from it have to be evaluated not just in relation to evidence but to the theoretical position which lies behind it and the adequacy of that position ...

To recap. In our model of argument an argument-1 is taken as the set of related propositions which make up the contents of a move in an argument exchange and is composed of a claim with supporting grounds. Grounds are related to claims by warrants or derivation rules derived from common sense or from the bodies of theory operating in the particular domain. The warrant is given a backing which may simply be a statement of provenance or authority or a set of higher level principles.

In order to deal with arguments from principles we make use of a simple model of a theory as a set of principles. A principle is a generalized statement of some belief held by the theory. For example it is a part of marxist theory that there there will be divisions within society. Such principles can be coded as frames which in turn have grounds and so on. Obviously at some point the sequence of grounding theories will have to end. Some grounds will have to be simply accepted as given whether by the theoretical viewpoint or by the common sense viewpoint which structures the individual's experience and hence argument. Indeed there are probably only a very few such ultimate grounds (mostly to do with harm to a person and his/her immediate family and so on³).

We can think of a viewpoint in a spatio-temporal sense. That is to say that two viewpoints may differ in the physical and temporal location of the observer. This is not really important here though it would be in a court case concerning say a street mugging. We can also take viewpoint in an ideological sense. Here I mean the set of views, opinions, beliefs and ideas which we all carry around inside our heads. We all have this and interpret / make decisions in its light. Further we are all capable of entertaining more than one viewpoint at once. I am indebted here to Roger Fowler's (1986) discussion of this in a literary criticism context. Indeed, the discussion of viewpoints in terms of the viewpoints that characters in literary texts have or convey has much of interest for the argumentation theorist. A field can be taken to mean a discipline such as archaeology or literary criticism. A viewpoint on a field is the set of warrants and so on which are used to interpret the field; that is, what the interpreter knows about the field or is experienced in. (A similar notion of viewpoints is suggested by Wilks and Bien, 1983). If we have a dynamic view of field such that what constitutes a field is determined by the collective purposes of the arguers within that field (Rowland 1982) then the purpose could be said to be *part* of the viewpoint or ideology. Thus we have the following marxist principle and its common sense ground:

```
principle::
  name: marxist_principle_1
  claim: cause(class divisions, conflict within groups)
  grounds: common_sense_principle_2
```

```
principle::
  name: common_sense_principle_2
  claim:
    cause(divisions between groups based on X,
           conflict within groups based on X)
  grounds: basic_ground
```

A warrant or rule based on these would be of the form:

```
W123
IF
  evidence_for class_division
```

³ Midgley (1980) is a useful attempt relate practical reasoning to our nature as animals.

THEN

evidence_for conflict_within_groups.

This warrant and others derived from one or more theoretical viewpoints would allow the construction of an interpretative argument. This argument can be attacked by finding a weak link or links and attacking these, by defending another claim and so on. Another way of attacking it is to take a warrant such as W123 above and attack the principle or principles on which the warrant is based. Thus the use of W123 allows the responder to attack the argument by attacking the principle which grounds one step in the interpretative argument. One advantage of the model of arguing outlined above is that it can provide this possible response, which I have called *level shift*. An argument, in my model, is generated by warrants derived from different theories including common sense theories. These theories can also provide a set of beliefs which can be used to justify a decision. Level shift occurs when the argument focus moves to a discussion of one or another of these theoretical positions. To give a concrete example: consider the debates generated by Martin Scorsese's film *The Last Temptation*. An argument about the merits (or demerits) of the film can focus on (see Figure 4.8):

- a) The facts of the film - (e.g., whether Christ is recorded in the gospels as having sex with Mary Magdalene).
- b) A warrant drawn from a particular backing - (e.g., whether Mary Magdalene and Christ having sex is blasphemous given the legal definition of blasphemy as the depiction of Christ in a manner likely to give offence).
- c) Completely within the backing theory - (e.g., an argument about the use of legal sanction to support theology, the theological reasons for blasphemy, sociological reasons for blasphemy and so on).

4.3.3 Inadequacies in this approach to analogy and principles

As Hodder points out, it may well be that relational analogies demand a theoretical framework for their proper application. He says, 'The use of relational analogies depends on a good theoretical framework within which one can identify what is relevant in a particular case.' (ibid). Thus there may be no plausible analogy without

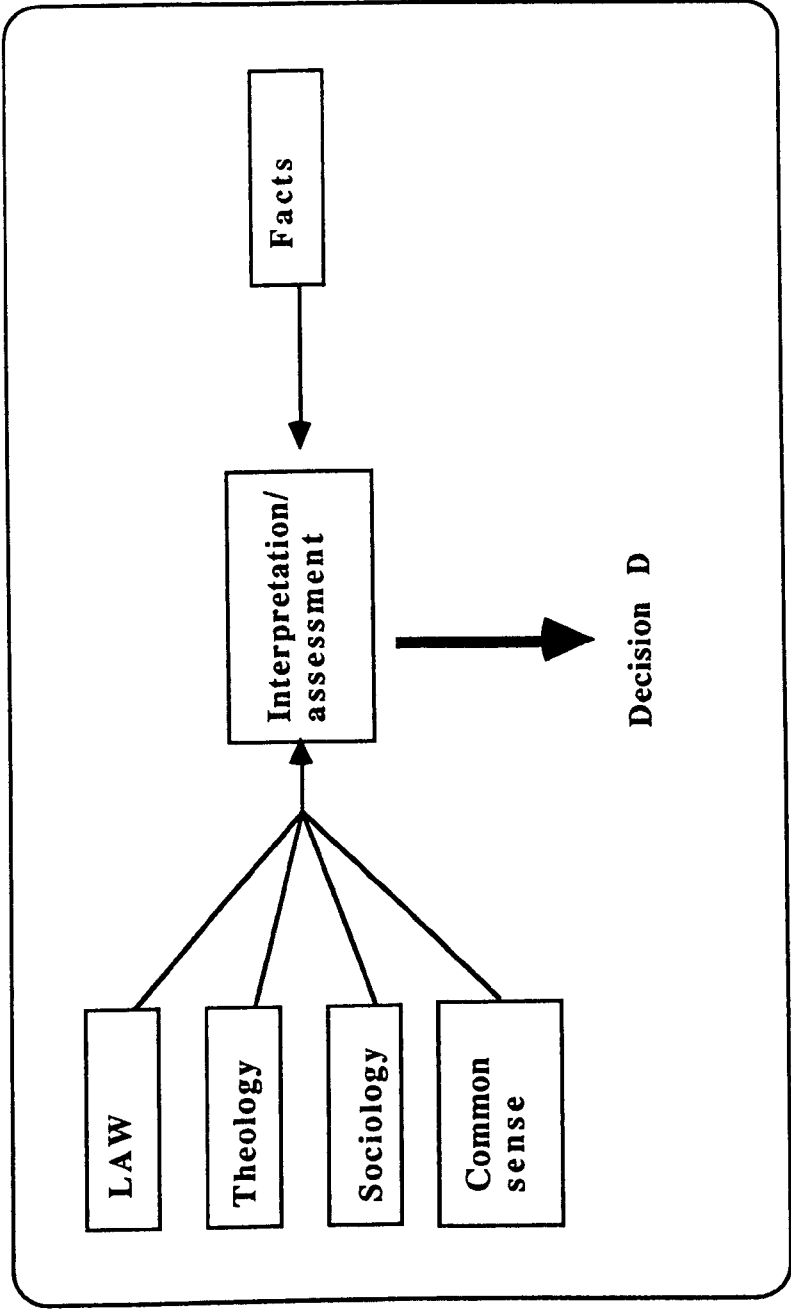


Figure 4.8 - Debate about The Last Temptation and its background

theory. Although I have made no attempt to show how analogies can make use of theories (models or principles) I have shown how a system is possible which makes use of both arguments from analogy and those from theory. There is no reason why our model should not be able to fully integrate the two styles of argument into a long chain in which an argument from analogy has as some earlier step an argument from theory. This sort of approach to analogy is similar to that of Keane (1985) and Kedar-Cabelli (1985) where analogies depend on the purpose of the maker of the analogy. But while the arguer's intention may be relevant, it is more important to identify the theoretical framework which can determine that the attributes of the base and target analogues are related within the base and the target in the appropriate way. This can only be given by some model or theory which constrains the analogizing process.

4.4 Relevance

In our attempt to characterize fields for argument we have so far dealt with one aspect of the problem - that of characterizing the operations which can be applied. An argument style is one means of specifying the types of low level reasoning operations which may be applied in a given field. However the context of a field is more than the style or operations. It includes the *relevant* facts as well as the particular examples of the reasoning types - the warrants or rules. We can think of the relevance of an argument as depending on a) the relevance of the operation types (i.e. style) and b) on the relevance of the facts operated on and rules which do the operating. The proper context for arguing is given by the relevant facts plus the relevant operations. In the previous section we dealt with relevance of operation types; in this section we will deal with relevance of facts and warrants. As in the case of style, it is important to be able to deal with relevance here since the machine will need to have some idea of what is relevant in producing a response. At the same time this knowledge will aid in the comprehension of the point of a user's argument. There are three types of relevance: immediate, indirect and tangential.

- Immediate relevance is the kind most easily dealt with in a knowledge base. The relevant facts are those which together with a warrant (or operation token) produce a particular proposition, end-state or goal. Thus if the goal is the answer to the question 'What is X?' then the immediately relevant facts are those which satisfy the conditional side of rules which conclude about X.

- Indirectly relevant facts and rules are those which need to be reached by a process of chaining to make the conclusion. The indirectly relevant facts, where the goal is as above, are facts which appear on the right hand side of a rule whose left hand side appears on the right hand side of another rule which concludes about X or which appears on the left hand side of a rule which indirectly concludes about X.

- Tangentially relevant facts and rules are those which compose an argument-1 which does not appear immediately or indirectly relevant to the topic at issue. Thus for example an argument-1 may address the topic of euthanasia by addressing the question of the suffering of AIDS patients. The latter is only made relevant as the result of some inference on the part of the other participant or as the result of a subsequent move on the arguer's part.

Relevance is an attribute of argument-0s which can be inherited by argument-1s. It doesn't make any sense to speak of argument-2s as relevant or not.

Relevance is also related to the notions of topic and level *shift* which have been mentioned above. Level shifts are relevant changes in the level of the argument from object level to the theory or models which back the object level warrants. The level shift is legitimate if a) there is a backing to some node which leads to the theory or model and b) this node is part of the current argument-2 (i.e in the argument net). It will also be legitimate if the node is immediately, indirectly or tangentially relevant to some node in the argument-2. Topic shifts are shifts from topic to topic. The relevant topics are those which are given by nodes on the argument-2 or which are relevant to nodes in the argument-2.

The notion of relevance shows how context goes beyond a set of facts and warrants grounded in theoretical principles (which constitute a viewpoint). The context grows as each participant adds to the argument. The overall context is made up of many sub-contexts (or many viewpoints on the facts).

There are of course other ways of dealing with relevance in systems of this kind. In his work on the use of critical argument in tutoring musical structures, Baker (1988) suggests a spreading activation model of relevance. 'Topics which are possibly relevant are those which are sufficiently *active*, in terms of spreading activation to other concepts in the semantic network.' (ibid:15)

Another method of dealing with relevance has been proposed by Sperber and Wilson (1987). In their discussion of communication they define communication in terms of contextual effects produced by the speaker by changing the hearer's cognitive environment (or the facts which a hearer is aware of or could become aware of). These contextual effects are the result of relating assumptions the hearer holds to new information bringing about changes in old assumptions or new assumptions. Relevance is defined in terms of this. 'An assumption is *relevant* in a context if, and only if, it has some contextual effect in that context' (ibid:702). The most relevant information is 'information likely to bring about the greatest improvement of knowledge at the smallest processing cost' (ibid:700). There is some unclarity about the notions of effort and effect adopted by Sperber and Wilson. For instance they say (ibid: 703):

We assume rather that the mind assesses its own efforts and their effects by monitoring physico-chemical changes in the brain. We argue then that effect and effort are nonrepresentational dimensions of mental processes: That is, they exist whether or not they are represented...

At the same time they suggest that relevance could be defined as a quantitative concept (ibid) presumably functioning in a manner similar to certainty factors. This would then depend on the values for effort and effect. Effort might be defined in terms of ease of access (see Sperber and Wilson, ibid: 703-4) to facts and rules (and would therefore depend on the particular architecture adopted) and effect would be a measure of the changes in the number of propositions believed (before and after) and the combined strength of belief in these (before and after). This is not incompatible with notions we are adopting and could be added so that X is the most relevant response if

- a) accessible as part of the current argument-2 i.e relevant at all, and
- b) it is most easily accessible (least effort) and produces the most effect.

Any system based on the model of argumentation given above needs some notion of relevance as well argument style since:

- a) It has to be able to select the operations which are best suited in a field to the production of *good* arguments and comprehension of/response to user arguments. This is given by argument style. This allows the system to attach to a field the more likely types of argument in that field. Thus producing arguments which fit the field and aiding processing of user arguments.
- b) It has to be able to select the facts and warrants for use in arguments - immediate, indirect or tangential.

To summarize, relevance is determined by the relations of facts and warrants to possible targets for attack or support. Warrant or fact X is relevant to topic T if there is an argument-0, A, which is about T and is in the current argument-2, and which can be supported or attacked by another argument-0 which contains X. Any argument-0 which is part of the current argument-2 is thus a relevant target for attack or defence. It will be a target for attack if it belongs to the arguer's opponent, defence if it belongs to the arguer.

Relevance is used for selection and comprehension. It is also used for *assessment*. In a system where there are meta-rules for assessment, the relevance of the arguments may be taken into account. Thus we could have the rule:

```
If
    Arg1 uses more relevant arguments than Arg2
Then
    Arg1 is better than Arg2
```

The kinds of relevance could be ordered {immediate, indirect, tangential} as a means of making the assessment. It is not clear however that the tangential relevance is least valuable. Often it is the point which becomes relevant after a couple of moves which is most telling. Thus the assessment rules would have to have some notion of *tellingness* as well as *relevance*.

4.5 Conclusion

In summary we have introduced a model of argument in terms of three levels. Interpretation is integrated into argumentation via level argument-1s. Various forms of reasoning are integrated at the argument-0 level. These include analogical reasoning which as we have seen is of the utmost importance to archaeology. It is also possible to integrate other modes of reasoning such as causal or other forms of plausible reasoning in this way (see Collins 1978). We have also introduced the notion of argument style as a means of distinguishing different domains in terms of the types of argument they employ and relevance to determine the appropriate warrant tokens. In the next chapter we shall discuss a computer program which has been designed in the light of these theoretical considerations.

Chapter 5 - Design principles and design

Argument: A set of design principles is introduced. In the light of these a program design is expressed, firstly, as a set of definitions and, secondly, in terms of the principal components both operational and representational of an arguer. In brief the argument process can be seen as the acting of an arguer on domain knowledge to produce an argument structure. In terms of the tripartite model, the procedural elements fulfill the role of the arguer, the data structures the role of representing the argument and the interpreter that of providing interpretations of the domain knowledge.

In this chapter I will provide a more detailed rationale for the the components of the program which embodies the ideas of chapter 4. The next chapter will deal with the actual details of the partial implementation of the model and provide example sessions using the system. This chapter will thus deal (a) with the functional architecture of the argument module and (b) with the structure of the underlying interpreter. I shall begin, however, with a brief preamble about the nature of models and a discussion of some of the principles which I have followed in designing the program and their rationale. I shall also include some semi-formal definitions which are embodied in the program design (see Figure 5.1).

5.1 Preamble about models in AI

Before I move on to a discussion of the system architecture I would like to say a few words about the thorny question of the relationship between programs, models and theories in AI. It is often naively supposed that that an implemented program represents a theory of some domain. This is perhaps because it is assumed that a theory can be defined as a set of propositions in terms of some formal system and a computer language is a formal system. However this will obviously not do. This is what we mean by a theory in logic but not in science (*pace* Hempel, 1968). A theory in science has to have some explanatory and/or predictive force (as Hempel, 1985, suggests). A model, on the other hand, is a set of objects (whether propositional or physical) which mirrors certain aspects of some real world object or system. As Moor

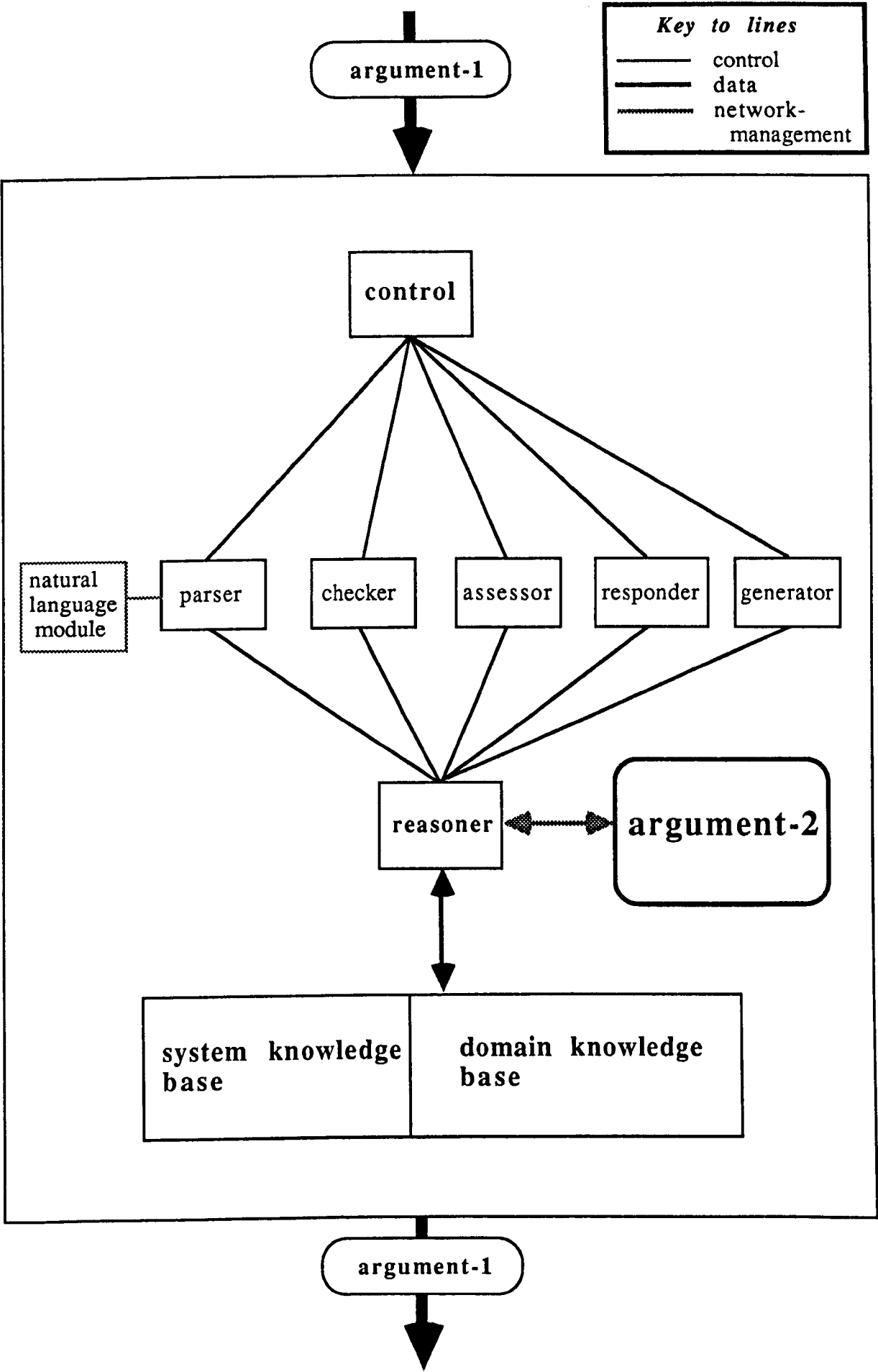


Figure 5.1 - The program design

points out (1978) , we can have a theory without a model and a model without a theory. Moor further points out that in order for a computer program to count as a model it must be given a '...symbolic interpretation which goes beyond the standard symbolic understanding of the the program in order to make it a model'. Moor makes use of Weizenbaum's ELIZA (1976) to show that a computer model may be based on entirely ad hoc implementational needs and cannot therefore constitute a theory of conversation.

It is important however that the programs we build in AI, like the bridges of civil engineers, are grounded in some theory of the domain they are concerned with. Thus we can see that we need some theory or model of argument in which to ground our program. Since, however, a theory is often stated in informal language, some computational model derived from the theory but taking account of what can be achieved computationally is required. The running program can then serve not as experimental evidence for the validity (or falsity) of the theory but as confirmation that it can partially redescrbed in terms of a programming language. This is related to, but does not go as far as, Thagard's notion (1982) of programs as '... simulations of models which approximate to theories' where a theory is not thought of as a set of sentences in some formal system but as 'a kind of definition' of some part of the natural world. In this view, a program could constitute a theory in the sense of a specification of some system. That this actually applies to human cognitive processes is a further claim which could be empirically tested. Because of the sorts of difficulties mentioned above Thagard resists the temptation to view programs in this way. He plumps for a notion of programs as simulations of models. A model is like a theory except that it claims only that there are analogies between the real world and the system. He concludes:

In sum, a program can not be said to be a theory or a model, but provides, when executed, a simulation of a system of a kind defined by a model which approximates to a theory.

In sections 5.4 and 5.5 I will present a design for a program which is partially implemented. The design and the partial implementation together provide a simulation of the system described in the previous chapter. To the extent that the design seems feasible and the program actually works the model put forward in chapter 4 is clarified, rendered more concrete and shown to have a degree of consistency.

5.2 Design principles

Apart from general design principles of efficiency and so on, the principles which act as constraints on the design are as follows:

P1: Since computers are good at the combination, permutation and matching of symbols but not at judging the results of these operations, designers of computer systems should attempt to produce assistants in the task of forming judgements rather than dogmatic judges.

P2: Any system which is to be used in the humanities must take account of the nature of the humanities.

P3: Any system which is to be used in the humanities must provide something by computational means which could not easily be provided in any other way.

These principles have been derived from many sources. In an essay entitled *Cybernetics and Ghosts* Italo Calvino makes the following observations (1986: 22):

The literature machine can perform all the permutations possible on a given material, but the poetic result will be the particular effect of one of these permutations on a man endowed with a consciousness and an unconscious, that is, an empirical and historical man. It will be the shock that occurs only if the writing machine is surrounded by the hidden ghosts of the individual and of his society.

The Calvino essay discusses a notion which has been widely canvassed ever since the birth of the computer: The computer as author. There have been countless attempts to produce poetry writing programs using various stochastic techniques. There have also been attempts to get the computer to produce stories. For instance, James Meehan produced a program in the seventies for the generation of fable-like narratives (Meehan, 1977). The Calvino quote expresses both why this has been felt to be a possibility and why it must ultimately fail. Since the machine is capable of tirelessly producing all the permutations of some set of basic data, if we could find a way of representing characters and situations, we could, for instance, encourage the machine to act as a scriptwriter for a TV soap. Of course there is more to the soap than the permutation of words or characters. Calvino suggests what this might be. That, in fact, the machine-author is not a human being with a human being's

sympathies, needs, desires, emotions, sense of humour, interests. Calvino suggests further that these are not something that can be codified since they are part of our lived experience. My feeling is that he is right in this view.

We can accept, then, that there are analogies between the products of writers and programmers and that artists, critics and other interpreters will go on using this technology both as a tool and as a source of metaphors. But as Calvino points out the machine has its limitations as creator. These limitations are not confined to the writing machine. All software machines are limited in what they can do. This is especially true of those which attempt to emulate or model aspects of human cognitive skills and is something that critics of AI have always seized upon. Hubert Dreyfus (Dreyfus and Dreyfus 1986) and John Searle (1980) are prominent among the philosophical critics. More recently, as we mentioned in chapter 4, Winograd and Flores (1986) have cast some doubt from within the discipline about the viability of the AI enterprise. They argue that since it is impossible to capture all the the background knowledge utilized by humans in solving problems it is better to use the computer not as an autonomous problem solver but as a kind of assistant which models only a subset of human skills. There are of course visionaries in AI who claim that the encoding of background knowledge is possible and therefore, by implication, that at least some of Calvino's ghosts can be confined within the machine. For instance, work has recently begun on the *Knoesphere* or CYC project (Lenat et al. 1983, 1986) to represent all the basic concepts necessary for the understanding of a standard encyclopaedia. Apart from Calvino, Winograd and so on mentioned above, similar points were made in an address to the 1985 Expert Systems conference by Margaret Boden.

The work I shall introduce below is an attempt to design a machine which can produce interpretations, criticize user interpretations and allow the user to criticize the system interpretations. In accord with the above principle, the system should make no authoritative pronouncements but should always be prepared to argue for its interpretations and criticism. Therefore, the project I describe should best be situated not in the domain of automatic inferencing (or expert systems) but in the domain of cooperative approaches to work (such as Winograd's 'coordinator', 1986).

Whatever the outcome of the debate within AI exemplified by Winograd and Lenat, I suggest that expert systems as oracles are largely inappropriate in the humanities. While in a domain such as oil exploration the user may require the

system to produce a single result (the most likely place to drill for oil) with perhaps some rationale for this result, in the humanities, we need to explore the many possible interpretations of our data. The difference in the needs of the two domains (oil exploration and the humanities) results from differences in their overall aims. The principal aim of oil explorers is to find oil and to exploit their findings. The principal aim of, say, the archaeologist is to add to the sum total of human knowledge. The role of the archaeological writer is not to hand down oracular judgments but to produce evidence or interpretations of evidence as part of the ongoing debate which constitutes the discipline of archaeology. Thus traditional expert systems, viewed as quasi-autonomous problem solvers, have only a minor role in the humanities and, if knowledge based systems are to fulfill a significant role, an attempt must be made to produce a design in accord with principles P2 and P3 (due, not to Calvino and Winograd, but to common sense insofar as any viable system in any domain must satisfy them).

As I have already suggested, I view the humanities as an arena (or set of arenas) where debate at conferences and through the medium of published papers is at least one means of advancing the body of knowledge which constitutes a domain. If this view of the humanities is correct, then it seems obvious that the sort of system we need to satisfy P2 is one which can participate in one or more phases of that process. The *ideal system* would be one which can store, display and allow single or multiple users to interact with vast bodies of data and opinion, which can provide and accept new interpretations of that data and argue in support of or against these interpretations. This is not something that can be done easily either by a human lecturer or by the use of, say, libraries and paper documents. The ideal system thus satisfies principle P3. At the same time, in accordance with P1, such a system should not impose its interpretations and evaluations, leaving the decision between competing interpretations to the user. In other words the system does not act as a decision-maker but as an aid to acquiring a considered view. The provision of multiple justified interpretations minimizes the risk of importing inappropriate interpretations, while imposing only a model of argumentation.

In the following sections we will present a design for a program which approximates to the ideal system outlined above.

5.3 Some definitions

The following definitions are derived from the model discussed in chapter 4. They represent aspects of the model in a well defined and hence computationally tractable form. In these definitions I have adopted the following conventions:

G = a ground or grounds

C = a claim

G \rightarrow C = G supports C

W = a warrant

G \rightarrow W \rightarrow C = G supports C using warrant W

variables are represented by uppercase initial letters (in Prolog style)

5.3.1 Debates, arguments, argument steps

A *debate* is composed of the interchange of argument-1s. An argument-1 is composed of argument steps or argument-0s and can form an interpretation. An argument step is composed of a claim, grounds and warrant.

A *debate* (or argument-2) is composed of a series of interchanges in which participants a) present views and b) defend and/or attack the views of other participants. In human debates each side will have a team of two members. One member from each team will talk for or against a proposal. The other member will then attack the opposing team's argument and/or defend the other member of his/her own team. The simplest form of debate is between two participants in which A puts forward a view, B puts forward a view, A counters, B counters. The debate is then assessed by a judge and a winner is declared. Usually in the Houses of Parliament, juries, local councils and radio debates this is determined by the voting of individuals. In the latter cases the assessment is, so to speak, distributed. The forms of debate possible with two participants A and B are therefore: ABAB-Assess, BABA-Assess, ABBA-Assess, BAAB-Assess.

A debate has a *field*. This determines the rules which (a) govern the production of arguments, (b) the response to arguments and (c) the assessment of arguments. A field has both contents (knowledge) and canons of good argument (or style).

A debate has a *topic*. This determines whether a particular contribution is relevant or not to the debate in process. A move is relevant if it is related though not remotely to the topic. The first move by A is always a proposition about the topic with support for this proposition. The first move by B is always a proposition relevant to

the topic plus support. This proposition will either be a direct contradiction i.e. it asserts the negation not(Proposition) or a contrary i.e it would be inconsistent to hold both Proposition1 and Proposition2.

An *argument-1* is composed of a series of propositions and their supports. These supports can in turn be composed of a set of propositions and their supports and so on.

Each *argument step* (or argument-0) is composed of a claim, grounds for this claim and a warrant which allows the derivation of the claim from the grounds. A warrant in turn has a backing which justifies the use of the warrant.

5.3.2 Moves in a debate

First moves have already been defined in the previous section. Second moves on either A or B's part can either:

- attack the first move of the other OR
- defend the first move of self OR
- restate the argument OR
- mixture of all three.

An *attack* on another's move can be either an attack on the grounds, the claim or the inference from grounds to claim. A *defence* of self's move can be support for the grounds, the claim or the inference. An attack on a claim is an assertion that not(Claim) or an assertion that Claim2 which is incompatible with Claim plus support for this assertion. An attack on a ground is:

- an assertion that not(Ground) OR
- an assertion that is incompatible with Ground plus support for this assertion.

An attack on an inference is an assertion that:

- not(Warrant) OR
- no support for Warrant OR
- not possible (in any viewpoint) to derive $G \rightarrow C$ using W or anything like W.

A support for a ground is the claim that Ground because Newsubgrounds. A support for a claim is another argument for the same claim. A support for an inference is:

- a support for a warrant OR
- a denial of attack on a warrant.

Typically support will be used for elements of A's argument which B attacks in B's

second move.

Response to arguments include attacks or supports on claims, grounds or inferences. Attacks/supports include direct attacks/supports, topic shifts, level shifts. A *topic shift* is a move to a (closely) related topic which has bearing on the topic under consideration. Thus for example the move from the morals of *Lady Chatterley's Lover* to the aesthetics of the text. A *level shift* is a move to a discussion of the grounds for the backing to a discussion of the principles which compose that backing. This can be illustrated by an argument about the film, *The Last Temptation*, which we have discussed already. Here a move can be made from a claim about the film, based on a warrant drawn from theology, to a discussion of theology.

An attack will be made on those elements which are found to be *weakest*. The weakness/strength of a step depends on its status. The *status* of an argument step depends on a) the success of the step, b) the type of the step and c) the argument field. An argument step is always evaluated in the context of a particular viewpoint and in the overall context of a particular field. A step *succeeds* if it is possible to get from the grounds to the claim using the warrants. Since warrants (and facts) make up viewpoints, the step is said to *succeed in a particular viewpoint* (VP) i.e. the one that contains the relevant warrant. This step may succeed:

in no other VP OR

in a VP with same warrant OR

in another VP with similar warrant.

A step also succeeds if there is a chain of steps which can be found to make the move from $G \rightarrow C$. The warrant (W) in this case may need to be expanded as well or it may be only applicable to one of the steps in the chain. Thus the expansion is to:

$G-W \rightarrow G_1 \dots G_n - W_n \rightarrow C$ (where W is used as the warrant in one step) OR

$G-W_1 \rightarrow G_1 \dots G_n - W_n \rightarrow C$ (where W is replaced by some new warrants).

The claim $G-W \rightarrow C$ succeeds if either of above is true. The status of the argument step depends on whether the step succeeds and how it succeeds. For example a step that succeeds by using an analogy will have status S1 in archaeology where S1 is high and status S2 in physics where S2 is very low because physics can make use of precise, quantifiable results.

5.3.3 Argument strength

The *strength* of an argument is determined by the statuses of its component steps. An argument is strong if it has no weak steps. An argument is bad if all the steps are weak. An argument is poor if the ratio of weak to ok steps is high. An argument is good if the ratio of weak to ok steps is low. The *overall assessment* of the strength of the arguments in a debate depends on the assessment of the combined strength of the two sides two arguments. As a computationally tractable first stab, the assessment of a participant's argument is given by the formulae:

$$F1: \text{overall_strength_side}(S) = \text{overall_strength}(\text{first_arg}) + \text{overall_strength}(\text{second_arg})$$

$$F2: \text{overall_strength}(A) = \text{strength}(A) + \text{strength}(\text{direct_supports}(A)) - \text{strength}(\text{direct_attacks}(A))$$

The overall assessment may be 'neither wins', 'A wins', 'B wins'. Depending on the the status of the overall assessment the participants will alter or leave unchanged their sets of beliefs including their beliefs about the beliefs of the other.

5.3.4 Argument types

The *type* of argument step is determined by the kinds of grounds and the sort of warrant. Thus an argument in a physics domain is differentiated from an argument in literary criticism both by the different evidence (bubble chamber tracks, poems) and the warrants used to interpret that evidence (laws of physics, common sense). The available types of argument step include simple rule based (or deduction), analogy, causal, induction, abduction and so on. It is possible to order these types in a precedence order. This ordering can be used both in assessing an argument and in deciding between the different types of argument to use. The precedence is related to the argument style. Argument step types are ranged in a default precedence *order* {simple-rule-based, causal, induction, abduction, relational-analogy, historical-analogy, simple-analogy}. This means that rule-based reasoning (or deduction) carries more value than causal and that carries more than analogical. This precedence ordering captures what I have called *mode based* argument style. *Level based* argument style (the appeal to models or principles as an explicit part of the argument) can also be included. Thus the full default ordering would be of the form: {simple-rule-based, causal, induction, abduction, relational-analogy, historical-analogy, simple-analogy, *appeal to models*, *appeal to principles*}.

5.3.5 Argument fields

The *field* is composed of the viewpoints (VPs) and the canons of good argument or style. The VPs are the facts plus warrants. The *canons* can be expressed as changes in the default ordering for argument types. Thus a possible precedence ordering for archaeology would be, {relational-analogy, historical-analogy, causal, simple-rule-based, *appeal to models*, *appeal to principles*, simple-analogy}. An argument may make use of a single VP or it may be composed of a series of argument steps embedded in different VPs; that is to say, that the argument steps which make it up may have warrants all drawn from the same VP or from different VPs. All arguing is done within a field or a domain. A field is not exactly the same as a discipline or the subdomains of that discipline but operationally we can take it as such. A field is a set of warrants, facts and canons. A topic marks out a specific part of the domain. The argument steps make use of the evidence or data from that sub-domain. There may be canons for each sub-domain. Knowledge sources (warrants) are used to reason about the domain data. These may be stored as a series of viewpoints. VPs may also include facts which are only facts within that viewpoint. A VP can also be a theory (where its components should be fairly structured), a personal VP, standard VP or a commonsense VP. These aspects of VPs are contained in the backing. Backing includes principles, models, simulations and commonsense.

5.3.6 Comments on the definitions

The above represents a stylized, formal and computationally tractable notion of debate and argument. Real debates are much less formal, more wide-ranging, less confined, value-ridden and emotionally charged affairs. Listen to the House of Commons or the radio program, 'Any Questions'. Real arguments are even more wide-ranging. There are few if any constraints on what can count as a claim or ground in an argument. The exchanges are extremely elliptical. Thus a single word or even action may contain a whole argument chain. We are adept at unpicking such chains. Arguments in general are compressed or elliptical versions of the above formal model. However, while the above is in many ways unrealistic, it has explicitly many of the elements of a real debate and even of a real argument. On the other hand the context-determined nature of real arguments, the common sense and shared background needed in interpreting and parsing the arguments of other, the wide ranging tracking skills, the keenness with which we detect aberrant moves in a fluid game are probably beyond the

capacities of today's technology. We would need at least natural language understanding and production, a complete range of human reasoning, empathy, emotion, values, shared backgrounds, common sense and an understanding of games and social interactions. We have all of these. Computers do not. Since mine is not an attempt to endow a computer with these but rather an attempt to get clear about some of the components of debate by modelling them on a computer these inadequacies are perhaps irrelevant. I shall have more to say about these matters in the final chapter.

5.4 The architecture of the argument system

The argument module is made up of a series of procedural components, which realize the model mentioned in earlier chapters and conform to the design principles above. The associated domain knowledge base and archaeological interpreter will be discussed in the next section. In what follows I will concentrate on what I regard as the central component of any argument system - the generation of the system's response to user arguments. I shall discuss the program under the following headings: procedural elements and representational elements. The former includes: argument checking; argument assessment; argument response; argument generation; control; reasoning. The latter includes: the argument node; the argument network; graphical displays; the system knowledge bases - strategies, trouble makers, theories. These are summarized in Table 5.1 which is an expansion of Table 4.2.

level in model	data structure	operations
=====		
argument-0	node/frame	inference forward chaining, analogizing slot access
argument-1	tree of nodes	parsing, checking, generation, response tree management
argument-2	network of trees	assessment network management

Table 5.1 - Data structures and operations for an ASP

5.4.1 Procedural elements

The principal procedural elements which correspond to the dynamic aspects of the model given in chapter 4 and which constitute the tasks which the *arguer* performs are dealt with in turn.

5.4.1.1 Argument checking

The checker takes as input the textual or other input of the user and attempts to confirm that this is either a viable argument as it stands or to produce a reconstruction of it which *is* viable. As a preliminary task the checker requires the parsing of any natural language input. In the implementation in chapter 6 the checker is realized as a set of rules for finding the weak status of nodes in an argument-1 tree.

Parsing

In the present design I have made little attempt to deal with natural language issues, preferring instead to concentrate on the model of argumentation. One way of reducing the problems raised by natural language issues is to use a window, menu and pointer interface. Another is the implementation of a simple Definite Clause Grammar. It should be feasible to add a natural language understanding module to the overall design (as indicated in Figure 5.1).

Checking

Argument step checking involves a process whereby the system must attempt to bridge any gap between the user's conclusion and his or her grounds. As Cohen (1987) has pointed out (see chapter 3 above), most arguments fail to state explicitly all of the arguer's premises. The system must therefore attempt to reconstruct the user's argument in cases where there is no warrant with which to derive a claim from a set of grounds. However there is one main difference between my work and that of Cohen insofar as I make much more use of analogy and other forms of non-deductive reasoning than she does. This is an implicit possibility in her work which I make it explicit. The basic job of the checker is to see if a parsed user argument holds. Since the system acts not solely as a deductive reasoner but also as a non-deductive reasoner, in order to determine this the system tries the following possibilities:

1. The claim follows directly from the grounds;
2. The claim follows from the grounds by an intermediate series of steps each of which makes use of a warrant given or found in the knowledge base;
3. The claim follows from the grounds + other grounds;
4. The claim follows from the grounds by analogy.

A component called the deriver can be used to see if claims follow from grounds. The deriver works by trying each of the possible means of getting to claim, C, from grounds, G, in turn, halting when one is satisfied or failing. The ways of getting from G to C may be ordered in the precedence order mentioned in section 5.3.4. Thus we could have analogy before the process of gap-filling given as 2 above. In cases where there is no complete chain between G and C, gap-filling is a three step process:

1. traversing the tree from the grounds as far as is possible, 2 traversing the tree from the claim as far as is possible and 3. seeing if it is rational to link the claim to the grounds at this point. It will be rational to link if the user believes a warrant which the system does not or the user believes in something which will allow the system to conclude that s/he believes in the warrant. All sorts of knowledge about the user and other users could be used to make assumptions about possible beliefs and hence to fill the gap between claim and grounds. This approach has a lot in common with Cohen's use of user models (discussed in chapter 3) and of user modelling work in general.

5.4.1.2 Argument assessment

In the preceding section we have dealt with how the system copes with checking individual argument steps. This can be thought of as a form of assessment of argument-0s and argument-1s. We now must turn to the overall assessment of an argument. There are many alternative methods. The problem here is that the easiest to implement - the counting method - is the least useful for an arguer since a system using it cannot produce reasons for its assessments. Thus in what follows I will outline the approach adopted and suggest alternatives which I would prefer but which are not yet possible.

In outline the assessor takes the overall argument and attempts to produce a relative evaluation. The simplest way to do this is to give some symbolic or numerical value to each component argument-1 and then to assess the overall value using the formulae given in section 5.3.3. According to these formulae the overall strength of an argument depends on the strength of the argument-1 itself combined with the strength of argument-1s which support it less the strength of those which attack it. The overall assessment (given in section 5.3.3 for a two move debate) would more generally be given by the total of all the strengths of component argument-1s. The evaluation of individual argument-1s can be produced in many ways. For example certainty factors may be attached to the rules used and the overall value of an argument-1 given by the certainty of its claim. In chapter 6 I will show how the evaluation of an argument-1 can be determined by taking the ratio of weak to strong nodes which make it up.

Other possible ways include the following. Using the method exemplified in the hypertext based system of VanLehn (1985) we could simply count the arguments for and against and display them in a matrix form like that given in Figure 3.1. A numerical method is also the approach favoured by Collins and Michalski (Baker, Burstein, and Collins, 1987, Collins and Michalski, to appear) in their work on plausible reasoning. Thus overall argument evaluation could be carried out by taking the graph structure contained in the argument network and counting the number of arguments and sub-arguments marshalled by each side which fulfill criteria such as accuracy, simplicity, parsimony, precision and generality. The main difficulty here is in producing formally defined accounts of these criteria.

Some ideas from the philosophy of science show how this may be possible. This work is aimed at finding a basis for the comparison of explanations. Since, as I have argued, arguments and explanations are closely related, this work provides a range of

possible candidates for argument assessment. Kitcher (1981) has suggested that a theory is better if it subsumes more facts. Thus Darwin's theory is better than any of its rivals because it can cope with more situations. However, in the end this comes down to simple counting. Furthermore, we can envisage a situation where an explanation does not cover as many facts as another but where it covers more significant facts. Work by Thagard (1978) has suggested a more satisfactory means of deciding between explanations. According to Thagard an explanation is better if it satisfies the following criteria: a) it is consilient i.e it explains a lot; b) it is simple; and, c) it makes use of analogies. More recently Thagard (1988a, 1988b, to appear) has advanced a connectionist model based on these ideas. In this model an explanation is assessed in terms of explanatory *coherence*. Using a set of weighting formulae represented in a connectionist program, his system, ECHO, can determine the best explanation. The system is based on a theory of *explanatory coherence* in which the best explanation is not simply the one which covers the most facts. Account has to be taken of how the various hypotheses are interconnected. Thus the coherence of an explanation depends on : (a) consilience - the number of facts (or pieces of evidence) which are explained ; (b) simplicity - the minimization of the number of hypothetical propositions; and, (c) the existence of links to other propositions in the set making up the explanation (for instance by relations of analogy). An explanation coheres better than another if the overall value of these taken together is greater. Since the calculation of this is not simply a matter of adding up the values (some tend to count against others) Thagard has opted for a connectionist implementation of the theory.

In order to allow for the production of reasons for an assessment, the best approach would be to take some of the insights derived from the philosophy of science and cast them in the form of a series of meta-rules. Thus the two arguments can be taken as objects with certain attributes and depending on these attributes they can be adjudged good or bad. For example a meta-rule for assessment might be of the form:

```
If
  arg1 subsumes more data than arg2
Then
  arg1 is better than arg2.
```

This rule is obviously an attempt to capture Thagard's consilience criteria. Other rules could be created for his other criteria. These might be added to by rules which were specific to the domain. Thus, following my discussion of analogy in archaeology in the

previous chapter, a rule which was of the form,

```
If
  arg1 contains a relational_analogy and
  arg2 contains a simple_analogy
Then
  arg1 is better than arg2.
```

is possible. We have seen already in the previous chapter how considerations of relevance might be used in assessment rules of this kind.

The main difficulty with this approach is that many of the standardly acceptable criteria for the acceptance of a scientific theory are not precisely characterized. Thagard has made an attempt at such a formalization. However as Hempel (1985: 121) points out, 'no exact and generally acknowledged formulation of corresponding criteria is presently available or likely to be forthcoming'. This suggests that there may be difficulties in producing other than the simplest meta-assessment rules.

5.4.1.3 Argument response

The responder takes the current state of the argument-2 and by selecting an appropriate component from the system knowledge base produces another argument-1 to add to the argument-2. The responder must take account not only of the user's argument but also of the particular argument style operative in the current field in order to determine which type of response is most appropriate in the particular case. The responder can be implemented either declaratively as a set of response rules or as a procedural module. In chapter 6 there are several examples which show how the latter option can be implemented.

As I suggested in the introductory chapter, in any automatic argumentation system it will be necessary for the system to have some means of responding to the user's arguments. In a reasonably sophisticated system where a range of different arguments are possible it is necessary to have a sophisticated means of response. To my mind it is the sophistication of response which is central to an arguer. In human arguments an argument-1 may be followed by another argument-1 in response which at first glance seems totally irrelevant. Yet over a period of time the relevance and cogency of the response will become obvious (that is, it is tangentially relevant as discussed in chapter 4). Thus the main difficulty in an automatic responder is in making sure that the response is relevant and as strong as possible.

The response which the system produces can be thought of as having been

produced as a result of a plan or strategy. However this term should be reserved for the overall strategy which is involved in highly manipulative persuasive argumentation. Characteristically the arguer here will be attempting to set a trap or to block his or her opponent's possible moves. The argument becomes, as it were, a game with all that that implies. The concept I am seeking here is more that which determines in a grammar-like manner what are the possible next moves and which is the best. This notion is similar to that we found in discussing the work of Reichman (see chapter 3 above). However I am unwilling to accept that the grammatical approach is best suited for an arguer. It does not seem likely that all of the possible moves can be encoded in advance. Even if this were possible it does not seem psychologically plausible. Nonetheless the approach has much to recommend it and I will offer a set of productions for response in a formal debate in chapter 6. The argument units of the Yale group also provide a means of analysing and responding to arguments. However these have a certain ad hoc feel. Thus while the approach of Reichman seems to be over-theoretical that of the Yale group seems under-determined. We need something which can stand as the basis for the system's response but which is grounded in empirical studies. There are two bodies of work which exhibit the necessary empirical basis and which are not over-determined: (a) The work of Perkins, Allen and Hafner (1982) on the analysis of everyday reasoning; and, (b) The work of Keane, Baird and Johnson-Laird (1988) on cognitive models of argumentation.

- In their paper Perkins et al. analyse responses to human arguments in terms of the use of mental models of situations rather than the committing of logical fallacies. As part of their account Perkins et al. give a summary of the kinds of counterarguments (they call them 'objections') which are used in everyday reasoning. In an analysis of about 2000 objections they found that eight types of objection accounted for 80% of the total. They call these objections the 'troublemakers'. I will give them in the order listed in the paper. In their discussion, relying as it does on a mental models approach they discuss the troublemakers in terms of causal models of situations. However, I feel that their categories can be applied more widely. Hence in what follows I have generalized from their causal line.

Contrary consequent. This is used when the same grounds are used to produce a different claim which is not consistent with the opponent's claim.

Contrary antecedent. In this case the claim of the opponent's argument is shown as following from a different set of grounds. Thus the opponent's argument is weakened since, while the claim may still be valid, there may be other reasons for that claim. Typically the responder will go on to claim that there are reasons for preferring the alternative grounds. This is not an attack on the claim but only on the means of arriving at it.

External factor. In this case an argument is refuted since some factor is present which affects the derivation of the claim from the grounds. In their example the claim that " We have a large population that would pull through in a military crisis" is objected to in the following manner: "A large population used to help, but today modern nuclear weapons can make short work even of a large population".

Disconnection. In this case the opponents grounds are perceived as having insufficient or no weight in support of the claim. In other words they are regarded as irrelevant.

Scalar insufficiency. This is Perkins et al.'s term for response to arguments which involve matters of degree. The arguer responds by showing that some factor is insufficiently present for the argument to follow.

Neglected critical distinction. Like the external factor response this involves the addition of some additional factor to the argument which allows a different claim. In this case the extra ground is that there is a critical distinction which is not made by the opponent but which allows a contrary claim.

Counterexample. This involves the falsification of some general claim or ground by the provision of counterexamples.

Alternative argument. This is the case where the arguer has put forward a claim and grounds which have been attacked by the opponent. The arguer's next move is

to produce an alternative argument for the same claim.

These troublemakers have the virtue of being empirically grounded. They are also clearly stated and thus reasonably easy to render in a computational form. On the other hand they are sometimes difficult to distinguish. For instance it is not easy to see the difference between *external factor* and *neglected critical distinction*. They are sometimes given over-specific definitions. For instance there is no reason why an *alternative argument* shouldn't be a response to an opponent rather than to an opponent's response. In this case *alternative argument* would act like *contrary consequent* and *contrary antecedent* except that a totally new argument for a claim relevant to the original is produced. Furthermore, two troublemakers may work together. For instance a *counterexample* is one way of showing that there is a *disconnection* between grounds and claim. In fact, the response is usually a two stage process in which the actual response is preceded by a process of discovering faults in the opponent's argument. Thus, in the case of *contrary consequence*, the contrary consequence is found and then this is used as the support for an argument attacking the opponent's original claim. If only the contrary consequence is given the argument will remain elliptical and require that the opponent infer the relation between the two arguments.

- Keane, Byrne and Johnson-Laird

On the basis of a series of experiments conducted at the Human Cognition Research Laboratory at the Open University, Keane has suggested that we can divide arguments into consequent, antecedent, negative-consequent, negative-antecedent, generalisation and specialisation forms.

A consequent type argument is one in which the logical or causal consequence of a proposition is derived, found to be either true or good and presented as evidence for the proposition. Thus

If the ballet is subsidised by the government

then it will be available to everyone

It should be available to everyone

Therefore, it should be subsidised.

This is the expanded form of

Proposal: Ballet should be subsidised

Argument: It should be available to everyone.

On the other hand, an antecedent argument argues for or against a proposal by looking for something which causally or logically implies the proposal and which is true or false (or good or bad). The negative forms of these take the negation of the proposal and use this in deriving either an antecedent or consequent which is either true or false (good or bad). A generalisation is an argument in which the response to a proposition is in terms of a generalisation which subsumes the original proposition. The example given is as follows:

Proposal: Ballet should be subsidised

Argument: All the Arts should be subsidised.

In my terms this represents the case where a claim is supported by an argument whose claim is the generalisation and whose grounds are unstated. A specialisation represents the case where a claim is answered by something which can be subsumed by the claim. The example given is

Proposal: Hanging should be made law for certain criminal offences

Argument: It should be made law for really violent crimes.

Perhaps in this case the 'argument' is less a response than a restatement of the proposal.

Keane et al. claim, firstly, that their research shows that there are a limited set of responses which arguers use; secondly, that more positive than negative arguments are produced; and, thirdly, specific proposals are met with specific arguments and similarly for general proposals. The first claim fits well with the point made by Perkins et al. that there are a limited number of generally used troublemakers. This bodes well for the computational model of arguing I am attempting to construct. The second claim suggests that certain sorts of argument will only be produced in certain circumstances; that there is a high level dictum which limits processing. This fits well with the notion of domain dependent argument styles which I have developed and suggests that we need to go further in characterizing not only whole domains but particular sets of circumstances which will call forth particular argument types. Keane et al. go on to suggest that convincingness may be a function of the amount of work required by a hearer. This final point is related to Sperber and Wilson's view of relevance discussed in chapter 4 and suggests that there may be easily accessible strategies for response. Thus the first choice when presented with a generalisation will be to produce a generalisation. More work needs to be done on what the

succeeding choices may be.

The work of Perkins et al. and Keane et al. has much in common with that of Allan Collins and his associates (Collins 1978, Baker, Burstein and Collins 1987, Collins and Michalski, to appear). This research is aimed at building a computer implementation of a model of human plausible reasoning. It is therefore intended to model reasoning rather than argument. But as we have seen above reasoning and arguing are closely related in that it takes reasoning to produce the argument-0s which make up argument-1s and ultimately argument-2s. Their system (the Plausible Reasoning Simulation System) is more closely related to our concerns in that it attempts to deal with the sort of reasoning humans actually use when knowledge is uncertain and incomplete. It includes the capacity to use different types of inference depending on the kinds of relations between the objects found in its memory. Thus if it finds mutual dependencies it will try functional analogies. Baker et al. mention four types of relations between memory contents and the questions asked of the system: generalization, specialization, similarity and dissimilarity. These are obviously similar to those of Keane et al. above. Thus if the system needs to reason about the flower type of England it can use more generalized knowledge about the flower type of Europe, more specific about that of Surrey, analogies between England and Holland or disanalogies between England and Brazil.

This approach goes some way towards a theory of response since it is grounded in empirical studies and is computationally tractable though relatively untested. The sorts of responses which can be generated by these are numerous. Perhaps not enough for a human argument but sufficiently close. At the same time they are general and not tied to particular domains (as those of the Yale group are - see chapter 3).

5.4.1.4 Argument generation

Arguments are generated using the responses to argument as input and outputting either textual or graphical forms of these. If the system is called upon to select an argument to start the ball rolling it will simply forward chain on the knowledge base and select an argument at the highest possible level (social organization or higher). The system will need to take account of the most appropriate types of argument for the current field. The argument is presented via toulmin structures or via canned text templates encoded in the trouble makers. Pruning is possible and perhaps necessary. System arguments should have the 'gappy' quality of user arguments. Argument

generation could be performed by transforming the proof traces provided by the rule interpreter. This could be achieved by using a set of predicates which (a) extract sub-arguments which have been used previously in exchanges with a particular user and (b) produce the best ordering for the argument step. The notion of argument as tree transformation using knowledge of the structure of actual arguments is based on Weiner's work on similar transformations of tree structures for explanation in BLAH (Weiner 1980, Goguen, Weiner and Linde 1983). The partial implementation simply produces graphical forms of the argument-1s produced during the reasoning process when generating a response.

5.4.1.5 Control

Ideally the control module in an ASP acts as a form of meta-level architecture. This notion derives from the work of Genesereth and Smith (1982) where they suggest that various generally applicable forms of reasoning in expert systems (such as forward chaining) can be encoded declaratively as a high level set of rules which are used to decide which object level rule is tried next. From this point of view the meta-level knowledge about arguments contained in the assessment rules and the troublemakers can be used as a meta-level of control which determines the actions of the domain interpreter. The main difficulty with this approach is that, while it is feasible and attractive, it is very inefficient. Thus it is more likely that control will be dealt with by use of an algorithm of the form presented in the next chapter.

5.4.1.6 Reasoner

The principal component of the reasoner is the interpreter to be discussed below. The reasoner could be used by the other components of the arguer as a central component for the manipulation of rules. Thus the various structures manipulated could be dealt with by the same reasoner using rules for assessment, response and so on. It is more likely that the reasoning will be distributed throughout the various modules. In the implementation discussed in chapter 6 the various reasoning tasks are implemented as procedural Prolog code with domain interpretation kept as a separate task.

5.4.2 Representational elements

The representational elements represent the arguer's storage structures. In part, these correspond to the argument of the tripartite model of chapter 4 with the domain

knowledge dealt with by the interpreter.

5.4.2.1 The argument node

There is no great originality in either of the following data structures. The work I am concerned with is less to do with finding new ways of representing than in finding new things to represent.

At the base of this system is the argument node. This is the representation which stores a step in an argument exchange or debate (an argument-0). Without this it would not be possible to store the system or user argument, check it, respond to it, display it either graphically or textually. At its simplest, as we have seen above in chapter 4 an argument can be seen in terms of some claim or conclusion supported by some grounds or evidence

$G_1, G_2, \dots, G_n \rightarrow C$

However, while this may be sufficient for logical or mathematical proofs and many everyday arguments, there are aspects of argument which are simply not captured by this. Primarily, as we have seen, the above notation does not capture the theoretical context in which the argument is taking place. Thus it is necessary to extend the simple notation above to deal with warrants and backing. Thus I have adopted a modified form of Toulmin's argument structures as described in chapter 4.

The argument node can be seen as a frame structure with slots for its various constituent parts. A typical frame would have the following slots and contents as a means of capturing the content of a typical argument (we will see in the next section that other slots are needed to deal with the network management).

```
type: argument node
owner: user
name: argnode12
subfield: millie's camp
claim: 'there is cooking at areal'
datum: 'there is a billie pole and lard bucket at areal'

warrant: 'if there is a billie pole and lard bucket at A then
          A is for cooking'
backing: [common_ sense, cree ethnography]
```

This node will be created when an argument is parsed. The various slots are filled (in a way similar to the work of McGuigan and Black 1986) during subsequent processing. Only the final slot - warrant - is problematical. In the case of an argument generated by the system, the warrant is the rule used to infer the claim. The case of an argument generated by the user is more difficult and requires more processing. The warrant can be produced in three ways:

- A suitable rule is found
- A suitable rule is constructed from the claim and grounds
- The current background theories are used to produce a new rule

The backing is the set of background principles which either ground the warrant or which are currently operative.

As I have suggested already, the argument node can be used not only to store the argument step but also as a means of generating textual and graphical versions of the argument. Textual versions simply fill the appropriate slots in an argument template with the contents of the claim and grounds slots of the argument node. (It is possible that a mode for particular users could be employed to restrict or admit access to the other slots.) Display is simply a matter of taking the various slots from the frame representation and mapping them onto the appropriate graphics variables.

5.4.2.2 The argument network

While the argument node provides the basic node for the storing and display of arguments - the atoms if you like - it is the argument structure which captures and can be used to display the overall ebb and flow of the argument. The argument network (see McGuigan and Black, 1986, Cohen, 1987, Alvarado et al., 1986) can be compared to a semantic net or to a hypertext (see Conklin, 1987a, 1987b). It basically consists of argument nodes which bear relations of support and attack to each other. Thus a claim such as *There is cooking at areal* (with its attendant justification) can form part of the support for another claim, for example, *Areal was used as a domestic area*. At the same time the original claim can serve as an attack on yet another claim such as *Areal was used as a refuse area*. If we number the first node N1, the second N2 and the third N3, this can be represented in terms which the system can use in reasoning about the state of the argument. Thus for example we can have the

following propositional representation of the argument-2:

```
supports(N1,N2) &  
attacks(N1,N3)
```

Again as in the case of the argument node the overall network can be displayed to the user as a set of such relationships or in a pictorial form.

In order to maintain this network the argument nodes require additional slots. Thus the following might be added to the example given in the previous section.

```
supports: argnode2  
attacks: argnode22  
supported_by: [argnode12, argnode34]  
attacked_by: []
```

The argument network has several interesting properties:

- It can be stored between uses by a particular user. This is accomplished by saving the nodes to a text file and recovering when needed. Thus the system can support more than one user or even more than one network for each user. Furthermore there is no reason why the system should not have more than one network in memory at the same time thus allowing the current argument to relate to previous arguments either of this user or of other users.
- The network stores the dependencies between the various arguments in terms of supports and supported_by, attacks and attacked_by links. The system as a whole acts as a form of reason maintenance system in Doyle's sense (Doyle 1979, 1980). The dependencies are necessary since, apart from the need to show the complete relationships between nodes, there is also a need to update the network when some fact has been shown to be wrong, some rule inadequate or some previous piece of argument subsequently proved false. Thus both the network and the domain knowledge base may be subject to the system's *changing its mind*.
- This network can also act as a sort of blackboard. One of the main difficulties in computational models of dialogue is a mechanism for dealing with *topic change* as discussed above. Reichman attempts to solve this by means of an ATN grammar. Given the nature of the data structure as described so far, it seems that the

simplest way of managing topic change is by marking the nodes with a status slot.

```
status: active
```

In this way the system can either move to another topic (by means of topic shift rules) by selecting an active or current topic. The system can also use these values as a means of deciding whether the user's move is legal. In general the system can use the network with its status slots as a means of keeping touch with the state of an argument. If a status slot contains the value *winner* then the argument is over. If it contains the value *loser* then this argument cannot be used in another argument and so on. If there are still *active* arguments then these must be dealt with (unless the user explicitly calls a halt to the argument). The system can be seen as taking values from other slots as a means of filling these status slots with *winner* or *closed*.

In summary then, the network which is produced as a result of the user system exchange is composed of a series of nodes. Each of these nodes is of the following form:

```
type: argument node
owner: user
name: argnode12
field: millie's camp
claim: 'there is cooking at areal'
datum: 'there is a billie pole and lard bucket at areal'
warrant: 'if there is a billie pole and lard bucket at A then
          A is for cooking'
backing: [common-sense, cree results]
supports: argnode2
attacks: argnode22
supported_by: [argnode12, argnode34]
attacked_by: []
status: active
```

The network discussed above acts as the argument-2 of the model. As we shall in the next chapter extra links are needed in each frame in order to deal with the argument-1s which are made up of argument-0s and compose argument-2s.

5.4.2.3 Graphical displays

It would be possible to conduct an argument exchange with an ASP entirely in terms of textual interchanges. While this is how I sometimes present argument exchanges there are several reasons why this is inconvenient:

- 1) Interactive arguments are not normally typed - this is too slow and unnatural
- 2) ASPs have limited natural language capabilities and must therefore rely on a very stereotypical and constrained form of input. Typing all of this would be time-consuming and lead to errors.
- 3) Modern computers are capable of interchanges which make use of various menus, graphical displays and pointing devices.
- 4) Users in archaeology are well used to various forms of graphical display containing data of many kinds. These range from maps to drawings of pots to reconstructions of houses.

For these reasons I suggest a predominantly graphical display for the interface with pull down menus and selection via the keyboard and the mouse. I shall describe in what follows the principal graphical displays and how the user interacts with them.

- Graphical toulmin structures

The basic form of the toulmin structure or argument node is as portrayed in Figure 3.2. This is derived from Toulmin et al. (1979) and Marshall (1987). Its principal use is as a means of displaying argument-0s. Figure 5.2 gives an example. However a series of hypertext like links could be made to other components such as the overall argument network and the preceding and succeeding argument-0s which, with the current argument-0, form an argument-1.

- The graphical argument-1

In its graphical form this is exemplified by the display given in Figure 5.3 (where *hfl* is the name of a rule which acts here as a warrant). In this case the argument-1 is composed of a single argument-0. Again this acts principally as a means of displaying information about the interpretations. As we shall see this is the main medium of communication for the implementation discussed in the next chapter.

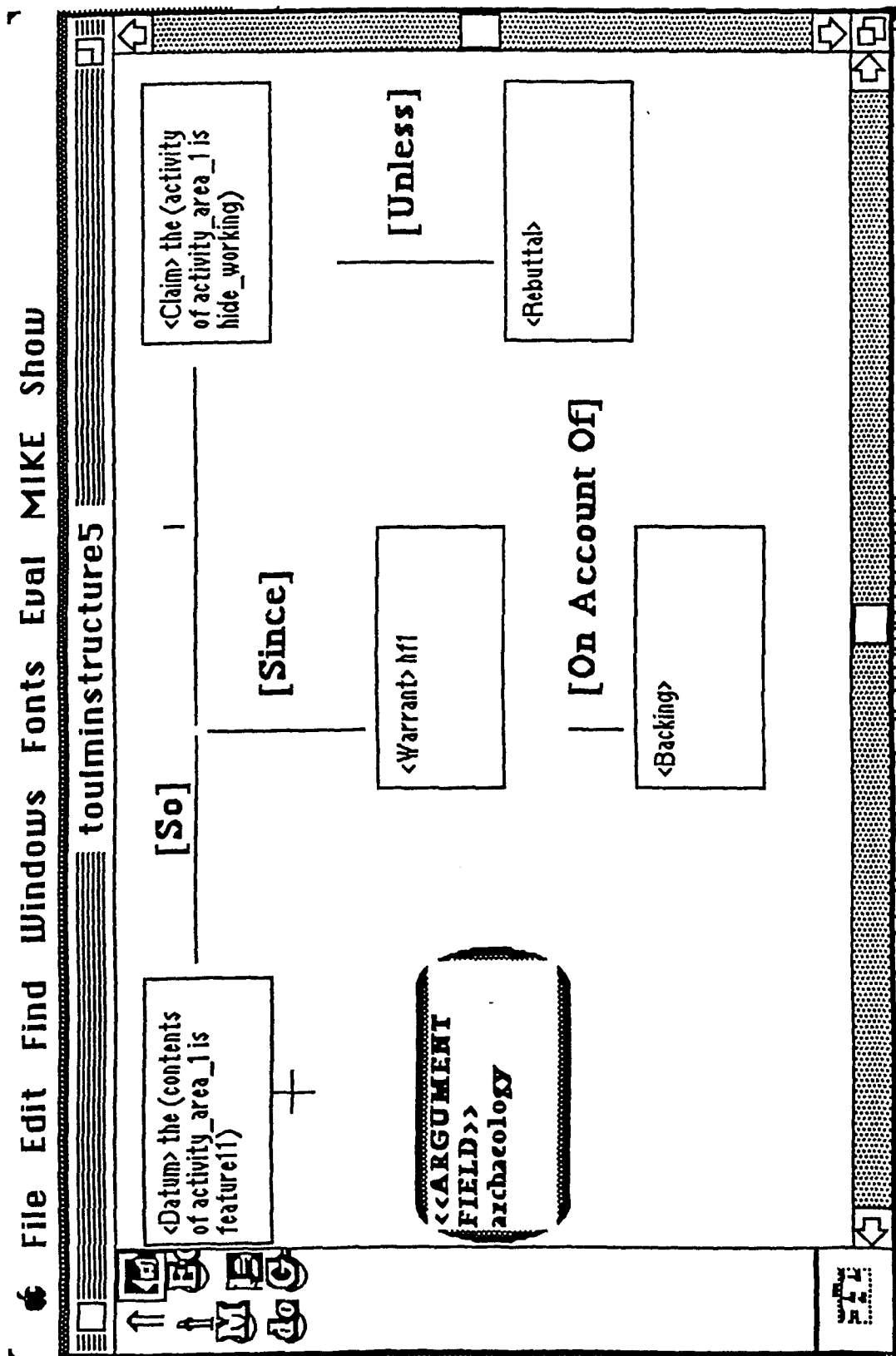


Figure 5.2 - The graphical argument-0 or toulmin structure

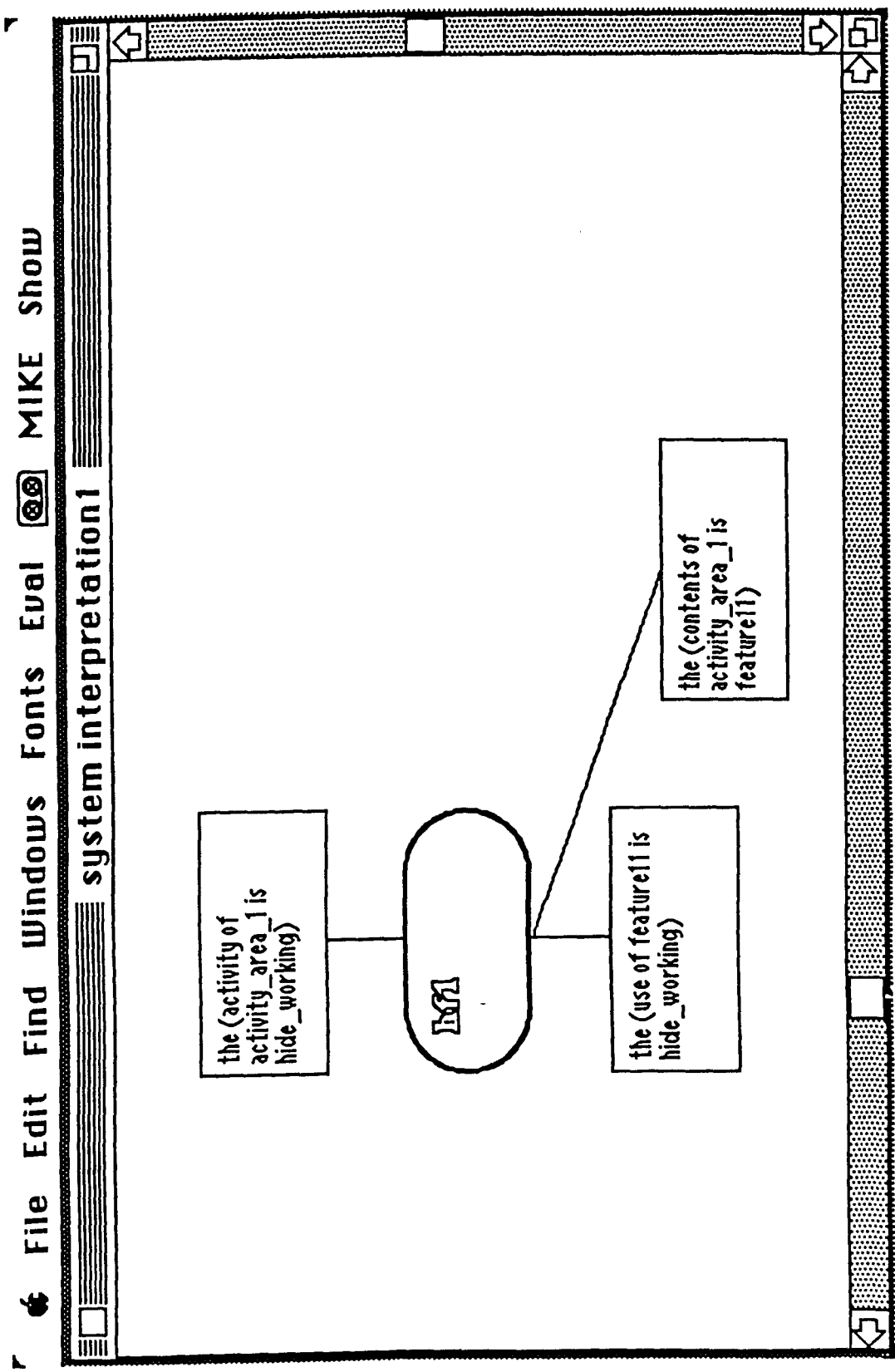


Figure 5.3 - The graphical argument-1 or interpretation

5.4.2.4 The system knowledge bases

The system makes use of a series of system knowledge bases during the various phases of the argumentation process. During parsing, checking and generation it needs a knowledge base of troublemakers, strategies and theories. During assessment a knowledge base of assessment knowledge is needed. During response, the troublemakers are again needed. By far the most developed of these at the present time is the knowledge base of troublemakers. I have discussed the origins and development of the knowledge used here in an earlier section, here the discussion will be restricted to the implementation.

In what follows I will discuss the troublemakers, strategies and theories.

(1) Troublemakers

I have discussed the background to these in section 5.4.1.3 above. As we have seen the troublemakers (TMs) can be used as a means of controlling the inferencing which takes place when responding to the user's argument. The extensions to the basic set given above are practical extensions aimed at facilitating various aspects of this. I would claim that they represent intuitively plausible responses to arguments but I have no empirical evidence for their existence. The full list of trouble makers is as follows:

contrary_consequence

contrary_antecedent

external_factor

disconnection

scalar_insufficiency

neglected_critical_distinction

counter_example

alternative_argument

reductio

ad hominem

response to analogy

topic_shift

level_shift

pick_up_the_thread

capitulate

An obvious extension to this set would be a group of eight troublemakers which dealt with the responses to the use of troublemakers. Given that a participant has employed one of the trouble-makers one likely response is to suggest that that trouble-maker was employed wrongly. This will be different for each trouble maker. Thus, for example, an appropriate response to the use of *disconnection* would be to show how the claim is connected to the grounds.

As I suggested above, experiments by Mark Keane at the Open University have suggested that a common strategy in arguments is to take a given view and reason forward from that to some consequence which can be shown to be 'bad' in some way. The following troublemakers can be used to take the user's claim and derive a contradiction with either the users beliefs or systems or other user's.

reductio

This trouble maker takes the user claim as a supposition and tries to derive a contradiction with either a) a mutually agreed proposition, (b) a proposition the user holds, or, (c) a proposition from some theoretical viewpoint.

pick up the thread

This TM looks for a currently active user argument node and sees if the argument can be continued from that point by further attacks.

capitulate

This TM represents the response of a rational arguer when s/he sees that the argument is lost i.e when there is no response possible.

ad hominem

This TM takes the user's argument and checks to see if any element of it contradicts something which the user has already claimed. This TM is thus a specific form of the *reductio* TM mentioned above. This is implemented in the prototype system and discussed as example 2 in chapter 6.

response to analogy

This TM responds as appropriate to the use of different types of analogies. Examples of this are given as example 1(a) and 1(b) in the next chapter.

topic shift

This TM takes an argument and failing to find an appropriate response otherwise shifts the focus of the argument to a relevant topic.

level shift

This TM has been discussed in the previous chapter. In essence a level shift is a topic shift where the new topic is at a higher level in the field. Thus the new topic is some part of a model or set of theoretical principles. This is exemplified as example 3 in chapter 6.

As I have suggested already, the troublemakers can be viewed as a means of meta-level control in the Genesereth and Smith (1982) mould. The most likely form of encoding is as meta-rules in a production system (Davis, 1980) but other representations are possible.

These troublemakers usually embody a two stage process in which the actual response is preceded by a process of discovering faults in the opponent's argument. While it would be possible to encode each of the troublemakers as, for example, a Prolog predicate, in the partial implementation of ASParch (Chapter 6) I have attempted to simulate the two stage process by having rules which find weak nodes in the argument structure and other rules which attack these weaknesses.

The TMs also need to have access to the knowledge about the argument style of the field and be able to assess the relevance of a claim. As I have suggested in section 5.3.5 the information about the field can be stored as a set of argument types and given a precedence ordering. Thus in the case of the *response to analogy* TM the system can use this information to deal appropriately with analogy. In the examples in chapter 6, the only domain the system knows about is archaeology and the precedence is encoded procedurally. The system also needs to be able to assess relevance since several of the TMs require knowledge of what claims are relevant to the argument currently being considered. These include *external_factor*, *neglected_critical_distinction* and *topic* and *level shift*

(2) Strategies

Strategies for arguing can be defined in terms of the notion of attack and defence. A *cooperative* strategy is one in which an arguer attacks and defends both the

arguments of its opponent and its own. An *uncooperative* strategy is one in which (as with most of us) we defend our own arguments and attack those of our opponents. An *altruistic* strategy is one in which we attack only our own arguments and defend our own and our opponents. A *saintly* strategy is one in which we attack only our own arguments and defend only those of our opponents (though the notion of an opponent becomes unclear here - we are our own opponent). These strategies operate at a higher level than the troublemakers. They thus represent a meta-meta-level of control.

(3) Theories

A theory can be implemented as a set of frame structures such as the following example:

```
principle(
    backing_set: historico_ecological,
    name: h_principle_1,
    type: explanatory,
    claim: cause(invasion,
                conflict_within_groups),
    grounds : common_sense_principle_1,
    strength: 8).
```

Each frame contains a single principle of the theory. Each principle is either explanatory or predictive. A theoretical principle is otherwise dealt with as an argument node with slots for claim and grounds. This representation allows level shift to occur so that the theory becomes the object of a discussion. The theory can also figure in explanations or justifications. More will be said about the use of theories in the next chapter.

5.5 The architecture of the interpreter

In accord with the ideas mentioned in chapter 4 the interpreter can be construed in terms of a set of facts and transformational rules which operate on these facts.

The architecture of the domain interpreter can be exemplified by our work on designing KIVA, an expert system for archaeological reasoning (Patel and Stutt 1988, 1989). In KIVA the knowledge base consists of facts about the domain and rules for interpreting those facts.

The main sub-classification of the facts is into *features* and *finds*. *Finds* can be

taken as the movable objects found on a site. These can further be decomposed into objects which are shaped for some human purpose - artifacts - and unworked, natural objects - ecofacts. *Features* refer to the physical components of the site and include pits, ditches, post-holes, rubbish dumps, mounds and so on. Facts about the site might also be divided in terms of the different sorts of evidence. Thus we might have evidence from the excavation of the site, historical evidence from documents which mention the site, experimental evidence where similar sites have been reconstructed and even inhabited, statistical evidence derived from an analysis of the artefacts found and so on. However, the principal focus has been on evidence from excavations which has not been analysed statistically.

The above represent the bottom level facts and their structure for archaeological reasoning. Since, as we have seen (chapter 4 above), archaeological reasoning can take place at many levels we must have some means of distinguishing between these levels. The basic levels at which reasoning can occur are activity areas, sites, groups of sites and cultures. There is a hierarchy of cultural activities which might be inferred from the material remains found at the site. Figure 5.4 shows the hierarchy of site facts and cultural activities. The basic process is one of mapping from the former to the latter and represents an instance of what Clancey (1985b) calls *heuristic classification*. This suggests that, while this interpreter is designed for archaeological interpretation and hence archaeological argumentation, it also conforms to a general class of interpreters.

Not all archaeologists would agree with these hierarchies but protocols from an archaeologist taken by my colleague Jitu Patel suggest that this is a fairly comprehensible approach to the organization of archaeological knowledge. In the interpretation the archaeologist takes the features and finds and first identifies activity areas and then assigns activities to them. For instance a hearth may lead to the assignment of cooking as an activity. In our work on Pueblo societies we identified four major types of activity area: living areas; storage areas; plazas or the out of doors places where activities occur; kivas or subterranean rooms used for ritual purposes.

In order to deal with these facts, features and activity areas the interpreter must have various types of rules. As suggested in chapter 4, the main components of archaeological interpretation are: classification; reconstruction; and, cultural interpretation. Thus a computational interpreter will have to have rules of at least these broad types. In addition there will be sub-divisions of these various types. For

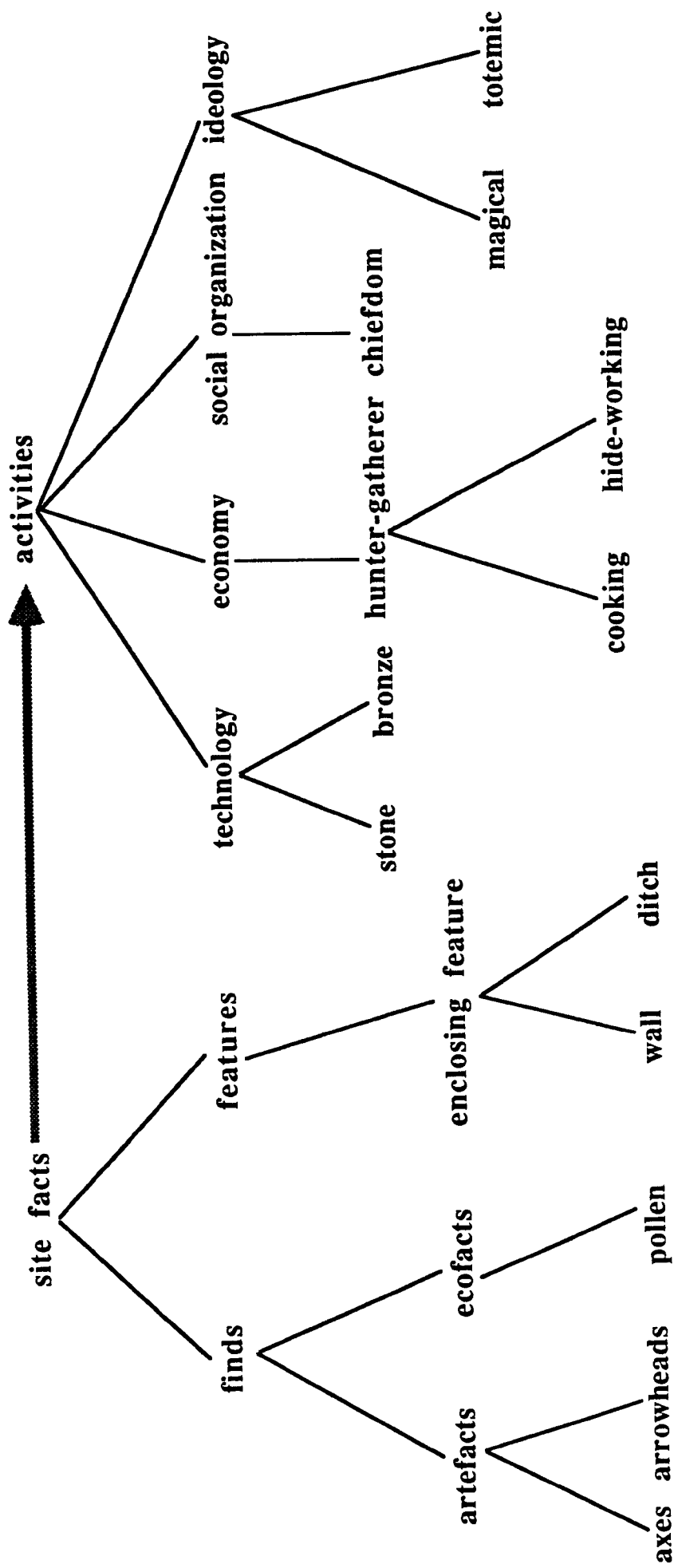


Figure 5.4 - The hierarchies involved in archaeological interpretation

instance, classification rules will be divided into rules for classifying features and those for classifying finds. In the KIVA implementation we had seven sets of rules. These were summarized as follows in Patel and Stutt 1988:

- 1) *features.rules*. This cluster of rules discovers significant areas from the size and placement attributes of features.
- 2) *finds.rules*. This cluster derives uses of artifacts from their attributes.
- 3) *content.rules*. This cluster takes individual areas and searches for *finds* and *features* within the area. It also checks that the contents are from the same period in time.
- 4) *areas.rules*. On the basis of contents, the main activity areas are identified.
- 5) *activity.rules*. From artifact uses, possible activities are inferred.
- 6) *constraint.rules*. Expectations derived from the model help prune the possible activities.
- 7) *site.rules*. Worlds are merged to give final interpretation of the site.

In terms of the model of archaeological interpretation portrayed above, rule-sets 1, 3 and 4 do some sort of classification; rule-sets 1, 2, 3 and 5 do reconstruction; and, rule-sets 6 and 7 attempt to provide cultural interpretation.

To make this more concrete I will include some of the rules. The following is an instance of the *features.rules* set:

```
if
    X instance_of hole &
    the lining of X is charcoal &
    the location of X is Loc &
    Y instance_of enclosing_feature &
    the border of Y is stone &
    the location of Y is Loc
then
    the type of Firepit is firepit &
    the location of Firepit is Loc.
```

The following is an instance of the *finds.rules* set:

```
if
    Artefact instance_of pottery &
    the base of Artefact is burned
then
```


the use of Artefact is cooking.

The following of the areas.rules set:

```
if
    Feature instance_of enclosing_feature &
    the placement of Feature is subterranean
then
    the type of Feature is kiva.
```

The following of the activity.rules set:

```
if
    Feature instance_of enclosing_feature &
    all enclosed_objects of Feature are Objects &
    the type of Feature is living_room &
    the use of Artefact is cooking &
    (member(Artefact, Objects))
then
    the activity of Feature is cooking.
```

The following of the constraint.rules set:

```
if
    the possible_activity of W is sleeping &
    the possible_activity of W2 is butchering
then
    the truth_value of W is false &
    the truth_value of W2 is false.
```

Since arguing involves several types of reasoning (as we have seen in chapter 4) the basic interpreter presented here needs to have extensions in order to deal with various forms of plausible and other reasoning which may be used by an arguer. There are many possible ways of doing this. For instance it would be possible to implement the search for analogies and so on in terms of procedural code. In our implementation (as we shall see below) we have extended the basic interpreter by including explicit rules for analogy finding. This represents another approach to the extensions required. It remains to be seen whether or not such explicit extensions are possible for other forms of reasoning.

5.6 Conclusion

To conclude, in this chapter we have produced an informal design for a program based on the model above. While we have not specified every component we have dealt with the principal procedural modules and the data structures these operate on. In the next chapter we will discuss one possible means of implementing this design. We have introduced a set of formal definitions which are an attempt to capture in definitional form the interactions between the various components. These definitions are, like the design, derived from the model given in chapter 4. We have also introduced a set of design principles which constrain the form that the design takes. Thus the design takes account of the nature of the humanities in two ways. Firstly, in that the humanities use argument the system accords with the humanities. Secondly, the system takes account of the particular form in which knowledge intended to form the basis for arguments must be encoded. The design obviously provides something in the humanities which could not be provided in any other way. No individual is likely to have access to the speed of computation, the storage facilities and the graphical display of a modern computer. Finally the system does not attempt to impose judgements. The principal aim of the system is to attack and defend arguments. The user is enabled, by means of the argument-2 structures produced, to decide more confidently between the various alternatives.

-o0o-

Chapter 6 - Implementation and examples

Argument: An implementation of the design is possible and is capable of dealing with realistic arguments. There are many extensions which can be made to the prototype system.

While the model presented above will cover many cases it is beyond the scope of this research to fully implement it. Thus, in order to lend some support to the claim that such a system is possible I have implemented a restricted version of the model. This makes no claim to being a fully releasable computer program. Indeed, as its main aim is to provide examples in support of my thesis it is far from that. Nonetheless it provides at least a hint of how such a program would appear while at the same time lending some support to my claims.

In order to render the program readily tractable I have confined myself to a model of formal debate. That this says anything about the full model of argumentation given above depends on the reasonable assumption that formal debate can be seen as a sub-set of informal argument and that each individual move in an argument can be equated with a move in a debate. On the other hand, the moves possible in a debate are much more restricted in that it is possible to tell in advance whose turn it is and what the intent of the move is. In short, a debate is a more formal version of an argument.

By formal debate is meant the sort of thing which goes on in debating societies and which is common in schools. In this two participants or two teams of participants are allowed a period of time in which to talk for or against a motion. At the end of the allotted period a judge (or in some cases, the audience) will decide who the winner is. Each team will have a first move in which their position pro or contra the motion is put forward and a second move in which further support is given to their first move or in which the first move of the other participant(s) is attacked. If we assign the letters A1 and A2 to one sides moves and B1 and B2 to the other's then the possible combinations are *A1B1A2B2-Judgment* and *B1A1B2A2-Judgment* as we saw in section 5.3 above. Given this confined scheme it is possible to implement a program in which the important issues are clarified. These issues centre on the notion of how in a given field an argument is assessed and a reasonable response is generated and

how analogy and theoretical argument are dealt with.

A further restriction is included in the implementation. In a participant's first or second move many separate (though related) claims could be made. In the program the user and system are only allowed one claim. Thus while in a real debate many parts of the opponent's argument will be attacked the ASParch program is limited to one attack. This is not a serious limitation since a simple algorithm could be implemented which responds to all the points in an argument.

This chapter discusses the implementation of ASParch and presents some examples which ASParch can deal with. In the final section inadequacies in the program are identified and future extensions are outlined. This is included here in a true Toulminian spirit in that arguments should take account of possible rebuttals.

6.1 The implementation of ASParch

This section has the following structure. I begin by presenting brief details of the underlying inference engine, present the basic algorithm for the debater and then discuss the various components which are involved, both representational (nodes, network, viewpoints, theories) and procedural (weakstatus rules including checking, response including generation, assessment, graphical display). The full Prolog code for the more important parts of the implementation is given as Appendix I. The program is implemented in MacPROLOG™ on an Apple Macintosh™ extended using the Prodigy-4™ board.

6.1.1 The inference engine

A full description of the basic reasoning mechanism used in ASParch can be found in the PD624 course text and reference manual from the Open University (Kahney 1989). The system - MIKE - is implemented in various versions of Prolog.

This interpreter is a frame- and rule-based system. In the following section, I will first describe in brief the frames and the forward chainer and then outline the extensions I have made to the system to allow for analogy.

- **Frames**

The basic frame used in MIKE has the following format:

```
Name instance_of <class> with
  Slot1: [Facet1: Facetvalue....]
```

```

Slot2: [Facet1: Facetvalue....]
Slot3: [Facet1: Facetvalue, Demon_facet1: Demon_code,....]
      .
      .

```

An actual example, taken from Kahney (ibid):

```

patient_1 instance_of patients with
    name: john_doe,
    eats: [fish, meat],
    symptoms:
        [value: [headache, spots, runny_nose, fever],
         inheritance: supersede,
         type: medical_symptoms,
         cardinality: any].

```

While this system allows the use of demons such as *if_added* and *if_needed*, ASParch does not make use of this facility. Nor does it make use of features such as cardinality and type. In fact even inheritance is rarely used.

- Forward chainer

The following is the syntax for legal rules in MIKE:

```

rule <rule-name> forward
    if
        <condition1> &
        <condition2> &
        ...
        <conditionN>
    then
        <action1> &
        <action2> &
        ...
        <actionN>

```

The conditions and actions can include access to the information held in frames as well as arbitrary prolog code. The actions can include the creation of new frames.

Frames are accessed via the following:

the <slot> of <object> is <filler>
all <slot> of <object> are <list-of-fillers>
<object> instance_of <class>
<class> subclass_of <class>.

Prolog code is indicated by giving it as the argument to the special form, *prolog*:

```
prolog(<prolog-goal>)  
prolog(<prolog-goal1>, <prolog-goal2>...)
```

New frames are created using the special form, *note*, in the following manner:

```
note <object> <instance-or-subclass-of> <class> with  
    <slot1>: <filler-or-list-of-fillers>, ...
```

This special form is also used to modify slot values:

```
note the <slot> of <object> is <filler-or-list-of-fillers>
```

As the system is implemented most of the reasoning required by ASParch involves only access to frame slots. The actual inference process is done once at the beginning of a session. This is required for efficiency reasons since, while MIKE is fairly efficient, it is still a demonstration program. Thus forward chaining of the rules is performed and ASParch relies on these to store all its required information.

- Extensions

In order for the MIKE interpreter to function as the reasoning mechanism for the ASParch program extensions to the deductive inferences possible were needed. Rather than change the basic MIKE mechanism I decided to implement analogical reasoning as a set of three rules for analogy. These correspond to the three types of analogy discussed in chapter 4. The rules make use of the information stored in frames to find analogies. The reader should bear in mind that I am principally interested in replacing unknown slot values (in particular *use* or *function*). Analogies depend on matches between slot names. This is crude but serves the present purpose of generating analogical arguments.

1. Simple analogy

A1 is a simple analogue of A2 if both have the same attributes.

```
rule analogy_rule_1 forward
  if
    start &
    prolog((has_attributes(Artefact1, List1),
              has_attributes(Artefact2, List2),
              same_atts(List1, List2),
              not Artefact1 = Artefact2))
  then
    the analogue of Artefact1 is
      [value: Artefact2,
       type: simple_analogy].
```

2. Historical analogy

A1 is a historical analogue of A2 if both have the same attributes and there is a historical link between A1 and A2.

```
rule analogy_rule_2 forward
  if
    prolog((has_attributes(Artefact1, List1),
              has_attributes(Artefact2, List2),
              linked_historically(Artefact1,
                                  Artefact2),
              same_atts(List1, List2),
              not Artefact1 = Artefact2))
  then
    the analogue of Artefact1 is
      [value: Artefact2,
       type: historical_analogy].
```

3. Relational analogy

A1 is a relational analogue of A2 if both have the same first and second order attributes (or relations such as *cause*) and the second order attributes govern the first order attributes. For example *cause* may govern the first order attributes *attracts* and *revolves_around* in the manner Gentner (1983) suggests (see chapter 4).

```

rule analogy_rule_3 forward
  if
    prolog((has_second_order_attributes
              (Artefact1, List1),
              has_second_order_attributes
              (Artefact2, List2),
              has_attributes(Artefact1, List3),
              has_attributes(Artefact2, List4),
              same_atts(List1, List2),
              governs(List1, List3),
              governs(List2, List4),
              not Artefact1 = Artefact2))
  then
    the analogue of Artefact1 is
      [value: Artefact2,
       type: relational_analogy].

```

From this it can be seen that in any derivation ASParch will make use of steps which might be deductive or analogical. Thus an argument can be composed of a mixture of types of steps.

While the MIKE system has its own explanation facility I have chosen to store the system's reasoning in the following manner. My basic rationale is that I think of explanation as a process whereby an object - in this case a proposition - is interrogated as to the warrant and grounds which allow it to be asserted. Standard expert system explanation would thus proceed by following the chain of grounds and claims. In order for this to be possible the system makes use of rules such as the following:

```

if
  Feature instance_of enclosing_feature &
  all enclosed_objects of Feature are List &
  the type of Firepit is firepit &
  prolog(member(Firepit, List)
...
then
  note(the type of Feature is
        [value: living_room,
         justification: living_room_rule,

```


grounds:

```
[Feature instance_of enclosing_feature
all enclosed_objects of Feature are List,
the type of Firepit is firepit,...])
```

By means of this rule the system stores *all* the justifications and grounds for all of the changes made in a frame. This can be exemplified by the following:

```
feature11 instance_of site_features with
  name: feature11,
  type:
    [value: living_room,
 justification: living_room_rule,
 grounds:
   [feature11 instance_of enclosing_feature,
    all enclosed_objects of feature11 are
      [firepit6, artifact2],
    the type of firepit6 is firepit,...]
...]
```

It is important to note that these justification and grounds slots are only used if either the user or the system makes use of the claim to which they are attached. Thus, seen in this light, a frame for an object in the domain knowledge base can be thought of as a congeries of propositions about that object. As the reader can see the individual claims about each attribute value in the frame are stored as the facets: value, justification and grounds.

6.1.2 The basic algorithm for ASParch

The code for the central parts of the system is given in Appendix I. In this section I intend to give a simplified version of the actual algorithm in Prolog-like form in order to present the bare bones of what the system does and how it does it. For those readers unfamiliar with Prolog syntax there are a number of introductory text books such as Bratko (1986). I make use of the popular Edinburgh syntax for Prolog. The important point is that Prolog operates as a logic programming language. As such a clause or rule either fails or succeeds. Thus in the dummy code

X:-

Y,

Z.

X will succeed if Y succeeds and Z succeeds. The ':-' sign is equivalent to a reversed If...then rule, with the conditions to the right of the sign. In this case - If Y and Z then X. In Prolog 'and' is represented by the sign ',' and 'or' by ';'. Variables are given in uppercase. A clause succeeds if it finds an appropriate fact in its database or if it finds another rule which succeeds. The Prolog interpreter thus works as a backward chainer performing a depth first exhaustive search.

In the succeeding sections I will discuss the central components of the system in greater detail. Basically the algorithm allows the user to request either the system-user-system-user (SUSU) or user-system-user-system (USUS) form of debate. If the former the system looks for an argument, displays it, requests the user's first argument, responds to this and then asks for the user's second argument. A similar procedure is followed for the USUS form. The pseudo code is fairly self-explanatory. Where necessary I have included comments (marked by %s) and referred the reader forward to the following sections.

```
maincontrol:-
```

```
    get_field(F),  
    get_topic(T),  
    decide_first_move_and_proceed,  
    see_if_assessment_needed.
```

```
%% User-System-User-System form of debate
```

```
decide_first_move_and_proceed:-
```

```
    yesno(['Do you want to go first?']), %% answer = yes  
    ask_for_user_argument,  
    display_user_argument,  
        see section 6.1.8  
    look_for_initial_system_argument_ususform,  
    display_system_argument,  
    ask_for_user_response,  
    display_user_argument,  
    decide_whether_to_attack_or_defend,  
    display_system_argument.
```

```

%% System-User-System-User
decide_first_move_and_proceed:-
    % answer = no
    look_for_initial_system_argument_susuform,
    display_system_argument,
    ask_for_user_argument,
    display_user_argument,
    decide_whether_to_attack_or_defend,
    display_system_argument,
    ask_for_user_response,
    display_user_argument.

ask_for_user_argument:-
    get_claim,
    get_grounds,
    store_user_argument_as_node,
        see section 6.1.3
    recursively_get_sub_arguments.

look_for_initial_system_argument_susuform:-
    forward_chain_on_domain_kb,
    current_topic(the Att of Obj),
    prove(the Att of Obj is Value1),
        %% call the MIKE slot retrieval mechanism
    convert_production_traces_to_argument_node.
        see section 6.1.3

look_for_initial_system_argument_ususform:-
    forward_chain_on_domain_kb,
    current_user_claim(the Att of Obj is Value),
    (prove(the Att of Obj is Value1);
    prove(the Att of Obj is not(Value))),
    convert_production_traces_to_argument_node.
        see section 6.1.3

ask_for_user_response:-
    get_response_type(Response_type),
    claim_or_grounds(Focus),
        % see if focus is claim or grounds
    ask_for_user_argument,
    manage_links_new_arg_node_to_arg_net(Response_type),
        see section 6.1.3

```

```

store_response_type(Focus).

%% If no weak nodes found then support the system argument
decide_whether_to_attack_or_defend:-
    get_user_argument(Arg),
    find_weak_spots_in_user_argument(Arg,Weakspots),
        see section 6.1.5
    Weakspots = [], %% none found
    defend_system_argument.

decide_whether_to_attack_or_defend:-
    get_user_argument(Arg),
    find_weak_spots_in_user_argument(Arg,Weakspots),
    attack_weak_spots_in_user_argument(Arg,Weakspots).

defend_system_argument:-
    find_alternative_arguments_with_same_claim;
    find_alternative_arguments_for_a_ground.

find_weak_spots_in_user_argument(Arg,Weakspots):-
    examine_each_node_for_weakness(Arg,Weakspots).

%% This clause recursively examines the roots node and all the
%% sub-nodes of the tree for an argument-1 and builds up a
%% list of lists with each of the sublists representing a
%% {node, status} pair.
examine_each_node_for_weakness(Node,
    [[Node, Status] | Restofstatuses]):-
    get_node_claim(Claim),
    get_node_grounds(Grounds),
    get_node_warrant(Warrant),
    weakstatus(Claim, Grounds, Warrant, Status),
    recursively_examine_grounds(Grounds, Restofstatuses).

attack_weak_spots_in_user_argument(Arg,[Weakspot |
    RestofWeakspots]):-
    attack_a_weakspot(Weakspot),
    recursively_attack_weak_spots(Arg, RestofWeakspots).

%% attacks on analogy depend on argument field
attack_a_weakspot(Arg, [Node, Status]):-
    Status = analogy_ground,

```

```

        see examples 1(a) and 1(b)
current_field(archaeology),
find_analogy_type(AType),
deal_with(Arg, Node, AType).

attack_a_weakspot(Arg, [Node, Status]):-
    deal_with(Arg, Node, Status).

%% 1. look for saliency failure see example 1(a)
deal_with(Arg, Node, simple_analogy):-
    get_node_claim(Node,Claim),
    Claim = (the analogue of A1 is A2),
    current_field(Field),
    check_for_unmatched_salient_attributes(Field,A1,A2),
    make_system_argument(not(Claim), saliency).

%% 2. Look for another contrary analogy see example 1(b)
deal_with(Arg, Node, simple_analogy):-
    get_node_claim(Node,Claim),
    Claim = (the analogue of A1 is A2),
    find_another_analogue(A1,A2,Anotheranalogue),
    current_field(Field),
    check_for_contrary_analogue(Field,Anotheranalogue,A2),
    make_system_argument(not(Claim), another_analogue).

%% 3. Look for selfcontradiction see example 2
deal_with(Arg, Node, arguable_principle(P)):-
    get_node_claim(Node,Claim),
    make_system_argument(not(Claim), selfcontradiction).

%% 4. Look for weak principle see example 3
deal_with(Arg, Node, arguable_principle(P)):-
    get_node_claim(Node,Claim),
    check_possible_level_shift,
    make_system_argument(not(Claim), weak_principle).

```

6.1.3 Nodes and network - storing and relating argument-0s

An argument-1 is represented as a series of linked frames each of which represents an argument-0. Each frame has the following structure:

```
Arg instance_of argument with
  claim: (the Att of Obj is Val),
  grounds: ListofGrounds,
  warrant: Warrant,
  supports: Argsupported,
  supported_by: Argsupporting
  external_supports: Node1,
  external_attacks: Node2,
  external_supported_by: Node3,
  external_attacked_by: Node4
```

The argument-0 is composed of the contents of the claim, grounds and warrant slots. The grounds slot contains a list of the supporting propositions. The supports and supported_by slots provide pointers to other argument nodes as a means of relating the argument-0 to the other steps in the argument-1. The external support and attack links provide access to the other argument-1s which are related as either supports or attacks as discussed above in chapter 5. They thus serve as links in the overall argument network or argument-2. Code is needed to transform the results of the system's forward chaining (where the slots of frames for features and finds and so on are manipulated) into the frames which represent the argument. The argument-2 structure is thus kept separate from the domain knowledge. The justification, grounds and value slots of the features are used in this process.

6.1.4 Viewpoints and theories

The system as implemented has two viewpoints in the sense of system and user knowledge bases. These are handled as separate sets of frames held in the predicates *sys_belief* and *user_belief*. The domain knowledge is held in the main knowledge base. Theories (or theoretical viewpoints) are held in frame like structures representing principles which are linkable to the warrants used in arguments. There will be further discussion of these in example 3 below.

6.1.5 Response - the weak status rules

As we shall see in the examples below the prototype system includes a set of rules which are used when searching the nodes which make up a user argument for any weak spots. I will deal with the rules used in the examples and which are therefore

reasonably tried and tested. These are intended to deal with analogies and arguments from theory as discussed above in chapter 4. Other rules which are more speculative will be discussed in section 6.3 below. These rules are closely related to the troublemakers introduced in chapter 5 above. There we pointed out that the troublemaker often can be broken down into a two-fold process whereby the arguer firstly finds some fault in the user's argument and then argues against it with another argument. The weakstatus rules represent an attempt to capture the first part of this. The second stage is performed by the *deal_with* clauses 1-4 shown in the above pseudo-code.

1. Analogy

The system uses a simple rule which finds an *analogy_ground* if the proposition which is the claim of that ground is of the form *the analogue of X is Y*.

```
%% analogy
weakstatus(P,G,W,analogy_ground):-
    P = (the analogue of X is Y).
```

Here, and in the remaining rules, P stands for the claim of an argument node (or argument-0), G stands for the set of grounds and W stands for the warrant which relates G to P. The fourth argument in the clause, '*analogy-ground*', is returned by the clause as the status of the node. In this rule the second and third arguments are not made use of. For instance, in example, 1a below:

P is bound to (*the analogue of feature11 is feature10*);
G is bound to an empty list (i.e. there is no ground given for this);
W is bound to an empty list.

2. Self-contradiction

The system returns the status *self_contradictory* if the user holds a belief which contradicts a belief which counts as a claim in an argument node making up an argument-1.

```
%% self-contradictory
weakstatus(P,G,W,selfcontradictory(X)):-
    assert_user_kb,
    user_belief(X),
    contrary(X,P).
```

The code for contrary is given by the following:

```
contrary(X,Y):-  
    X = not(Y) .
```

```
contrary(X,Y):-  
    X = (the Att of Obj is Val1),  
    Y = (the Att of Obj is Val2) .
```

3. Arguable principle

Here the system constructs a warrant from the grounds G and claim P and uses that to access a set of backing principles. The lowest valued principle is then returned and becomes the focus for attacks. This is fully illustrated in example 3 below.

```
%% arguable principle  
weakstatus(P,G,DummyW,arguable_principle(Pr)):-  
    not G = [],  
    G = [G1],  
    W = (G1 -> P),  
    backing_for(W,B),  
    min_valued_principle(B,Pr) .
```

This rule checks that the ground is not empty, then constructs a warrant from one of the grounds (in these examples, there is only one ground) and looks for the principle which provides backing for this warrant and which has the lowest value. The bindings for the variables is as above except that Pr will be bound to the principle which is found. In example 3 below, Pr is bound to *(cause(power, show_of_prestige))*.

6.1.6 Response - control rules

In the implemented program the rules which control what sort of response is made are embedded in the procedural code give above as *decide_first_move_and_proceed*, *decide_whether_to_attack_or_defend* and so on. These are not implemented as rules in a knowledge base but they could be (if a more efficient interpreter was used). In this section I collect together and present in one place the distributed knowledge of what the next move should be. In these rules S1 and S2 refer to the system's moves, U1 and U2 to the user's moves. E refers to an element in a user argument (a particular argument-0 in other words).

If
 no move made and
 the form is SUSU
Then
 the system finds an argument S1

If
 no move made and
 the form is USUS
Then
 the user provides an argument U1

If
 form SUSU and S1 provided
Then
 user provides U1

If
 form USUS and U1 provided
Then
 system provides S1

If
 SUSU and S1 given and U1 given and
 element E of U1 weak
Then
 S2 = attack E or
 S2 = defend S1

If
 SUSU and S1 given and U1 given and S2 given and
 S2 = attack E of U1
Then
 U2 = defend E or
 U2 = attack S2

If
 SUSU and S1 given and U1 given and S2 given and
 S2 = defend S1
Then
 U2 = attack S2 or
 U2 = attack S1 or
 U2 = defend U1

If
 USUS and U1 given and S1 given and
 element E of S1 weak

```

Then
  U2 = attack E or
  U2 = defend U1

If
  USUS and U1 given and S1 given and U2 given and
  U2 = attack E of S1
Then
  S2 = defend E or
  S2 = attack U2

If
  USUS and U1 given and S1 given and U2 given and
  U2 = defend U1
Then
  S2 = attack U2 or
  S2 = attack U1 or
  S2 = defend S1

```

6.1.7 Assessment

A simple form of assessment has been implemented in the program. This implements the formulae given in section 5.3 above:

F1: overall_strength_side(S) = overall_strength(first_arg) + overall_strength(second_arg)
F2: overall_strength(A) = strength(A) + strength(direct_supports(A)) - strength(direct_attacks(A))

The algorithm for this is given below. It must be borne in mind that tests on this algorithm are limited to the confined range of example arguments which ASParch is designed to simulate. Appendix IV tabulates the results of using this algorithm on the examples given later in this chapter.

```

overall_assess(Value):-
  get_user_root(R1), %% i.e. user's first argument
  get_user_aux(R2),  %% i.e. user's second argument
  assess_argument(R1, Value1),
  assess_argument(R2, Value2),
  get_sys_root(R3),  %% i.e. system's first argument
  get_sys_aux(R4),   %% i.e. system's second argument
  assess_argument(R3, Value3),

```

```

    assess_argument(R4, Value4),
    combine_overall_values(Value1, Value2, OValue1),
    combine_overall_values(Value3, Value4, OValue2),
    winner(OValue1, OValue2, Value).

```

```

assess_argument(Arg, Value):-
    assess_root(Arg, Value1),
    get_directly_attacked_by(Arg, External_attacked_by),
    get_directly_supported_by(Arg, External_supported_by),
    assess_attacked_by(Arg, External_attacked_by, Value2),
    %% recursively assess this root
    assess_supported_by(Arg, External_supported_by, Value3),
    %% recursively assess this root
    combine_values(Value1, Value2, Value3, Value).

```

```

assess_root(Arg, Value):-
    get_total_nodes(Arg, Nodecount),
    get_weak_nodes(Arg, Weaknodecount),
    check_ratio(Nodecount, Weaknodecount, Value).

```

%% If both same then argument full of weak nodes

```

check_ratio(N,W,V):-
    N = W,
    V is 0.

```

%% If weakcount is greater than half possible nodes then poor

```

check_ratio(N,W,V):-
    N1 is (N/2),
    W > N1,
    V is 3.

```

%% If weakcount is equal to half possible nodes then middling

```

check_ratio(N,W,V):-
    N1 is (N/2),
    N1 = W,
    V is 5.

```

%% If weakcount is less than half possible nodes then
reasonable

```

check_ratio(N,W,V):-
    N1 is (N/2),
    N1 > W,

```

```

V is 7.

%% If weakcount is 0 then strong
check_ratio(N,W,V):-
    W = 0,
    V is 10.

%% Formula F1 above
combine_overall_values(V1, V2, OV):-
    OV is V1 + V2.

%% Formula F2 above
combine_values(V1, V2, V3, V):-
    V is (V1 + V3) - V2.

%% If the the value of the user's combined argument-1s is
%% greater then the winner is the user
winner(OValue1, OValue2, Value):-
    max(OValue1, OValue2, OValue1),
    Value = user.

%% else the winner is the system
winner(OValue1, OValue2, system).

```

6.1.8 Graphical display

In ASParch I have provided graphical representations of the user and system argument-1s. In the present implementation argument-1s are represented as trees. See Figure 5.3. In this I have attempted to show the different parts of the argument (Claim, grounds, warrant) graphically. I have also attempted to show the different kinds of propositions which make up the grounds. Argument-0s can be displayed as graphical toulmin structures (see Figure 5.2).

6.2 The examples

The three examples illustrate: (a) how ASParch deals with analogies; (b) how the theory level and the domain level interact; and, (c) how viewpoints are utilized. More specifically they illustrate the operation of the *weakstatus* rules and the use of the *deal_with* clauses discussed above.

Examples 1(a) and 1(b)

In these two related examples the system responds to the user argument by first finding an analogy as one of the grounds and then dealing with that analogy. To find the analogy the system inspects each node in the user's arguments U1 and U2 and checks for weak spots using the rules outlined above. In this case the system notes that an analogy is being used at a particular node. This is reported back and forms the basis for the second step - the selection of a response. The system as implemented discovers the analogy simply because of the use of the key word *analogue* in the ground in question. In order to respond to the analogy the system first checks to see what type of analogy it is. Given that the field is archaeology the system knows that certain types of analogy are appropriate while others are not (see chapter 4 above). Thus if the analogy is a simple analogy the system will attack it purely on that basis. If the analogy is historical or relational the system needs to find a more appropriate response as outlined above. The system as implemented only deals with simple analogies. In example 1(a) the system finds that there is a salient feature which any analogue of feature11 must have. Since this is missing the analogy fails. This is reported to the user. In the second example 1(b) the system discovers an alternative analogy. (See chapter 4 above for a discussion of the appropriate response to analogy use in arguments).

Examples 1(a) and 1(b) are based on a knowledge base based loosely around the following passage from Bonnicksen (1973) which was mentioned in chapter 1:

Activity area 7 occurs north and west of activity area 6, on the slope leading down into the gully. Two dead poplar trees leaning at approximately a 45° angle are supported by two live poplar trees at their top and a cross brace at the bottom. Hair from a large game animal, perhaps an elk, is scattered along the bottom edge of the frame. The frame construction is similar in shape and size to frames used in contemporary Cree camps for hide stretching.

Here Bonnicksen is using a simple analogy. This chunk of interpretation could conceivably be coded as a single knowledge base rule. However it seems more general to decompose into a set of rules. One will conclude about game animals from the existence of animal hair. Another about the existence of a structure from the existence of poles plus cross brace. Finally a rule will argue (by analogy) that these show the use as hide-stretching. S1 will be analogous to S2 if S1 has received an

analogy value S2 of type simple, historical or relational.

To avoid unnecessary complications at this stage, the knowledge base for examples 1(a) and 1(b) is as follows. The knowledge base for example 2 contains a fuller representation of the reasoning in the above quote.

```
%% FRAMES

feature11 instance_of structure with
    height: 2,
    width: 2,
    angle: 45,
    use: hide_working.

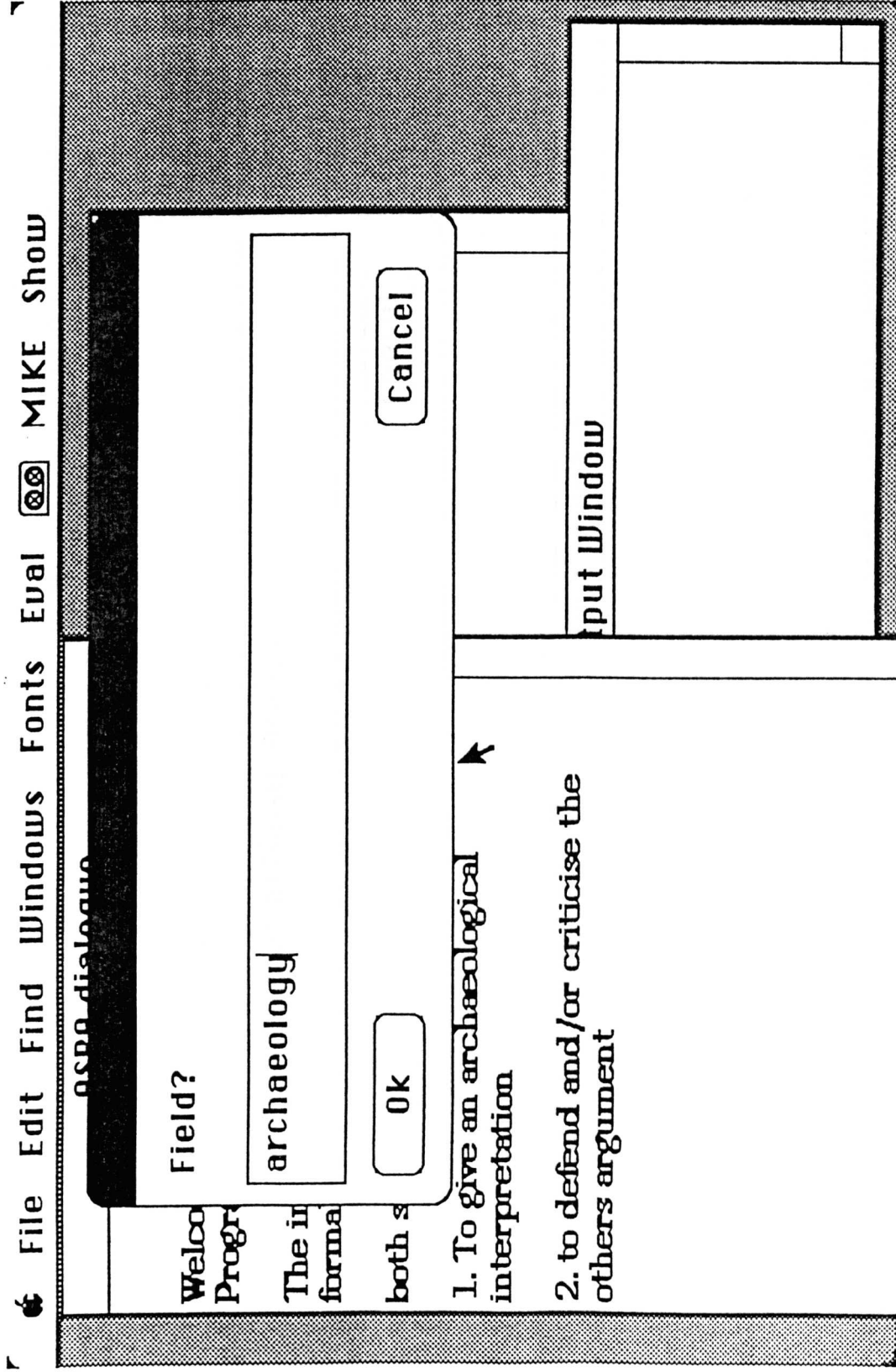
activity_area_1 instance_of activity_areas with
    centroid: [10,10],
    contents: feature11,
    discovery_at: deer_hair,
    activity: unknown.

% Rules
rule hf1 forward
    if
        the contents of AA is F &
        the use of F is Use
    then
        the activity of AA is
            [value : Use,
             justification: hf1,
             grounds:
                [the contents of AA is F,
                 the use of F is Use]].
```

Example 1(a) - dealing with analogy by finding a missing salient attribute

This example will be followed through in a fair degree of detail to illustrate both the specific point of how the analogy is dealt with and also generally to show how a full system could work as based on this partial prototype.

When the program is run the user first meets the screen shown Figure 6.1 where the user is requested for the field of the argument. Since the prototype only knows about archaeology this is the only sensible designation. In Figure 6.2 the user is requested for a topic. This necessary element of an argument is rarely given explicitly



Figures 6.1 - 6.10 - Example 1(a) - Macintosh™ output screens

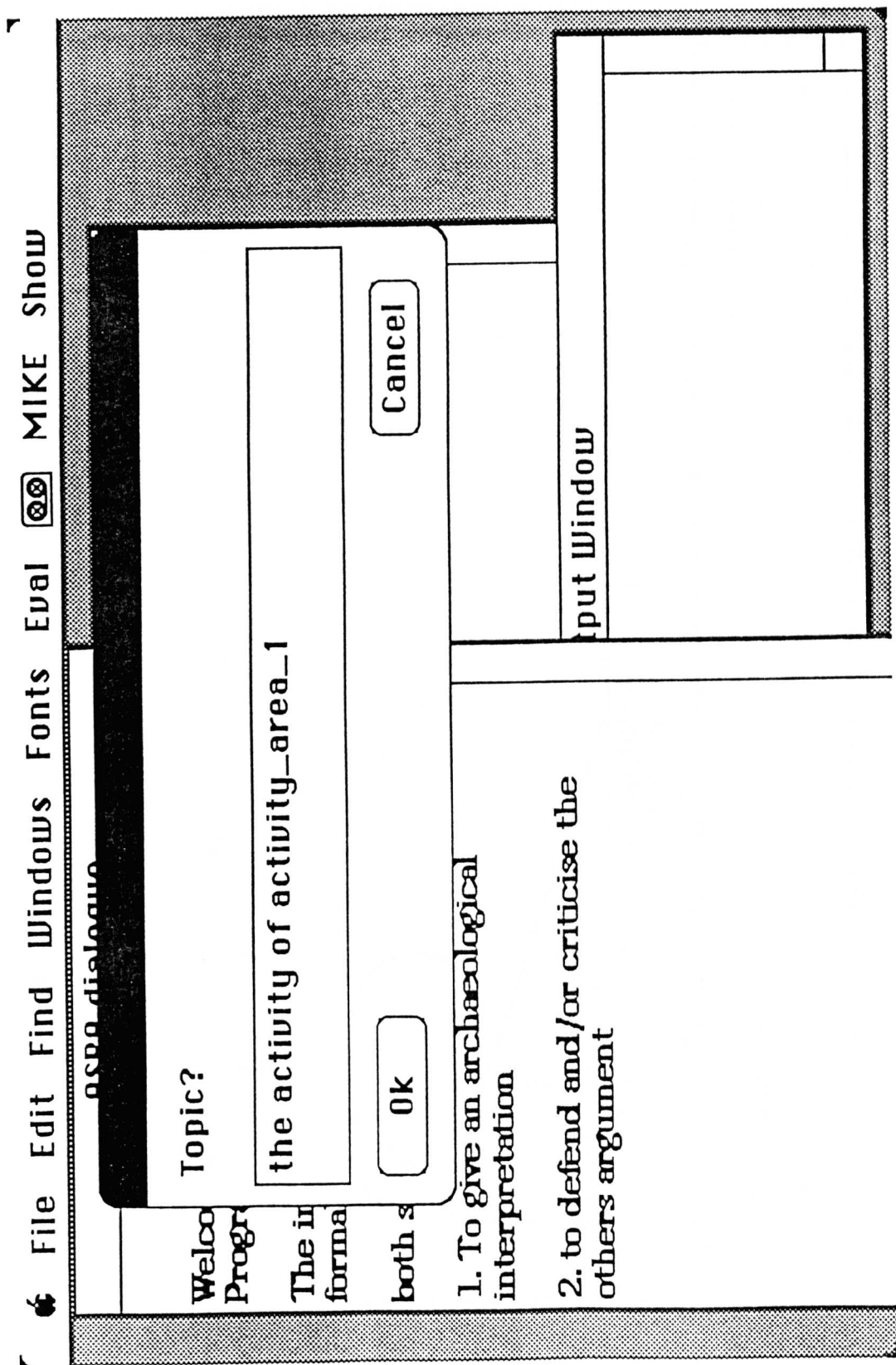


Figure 6.2

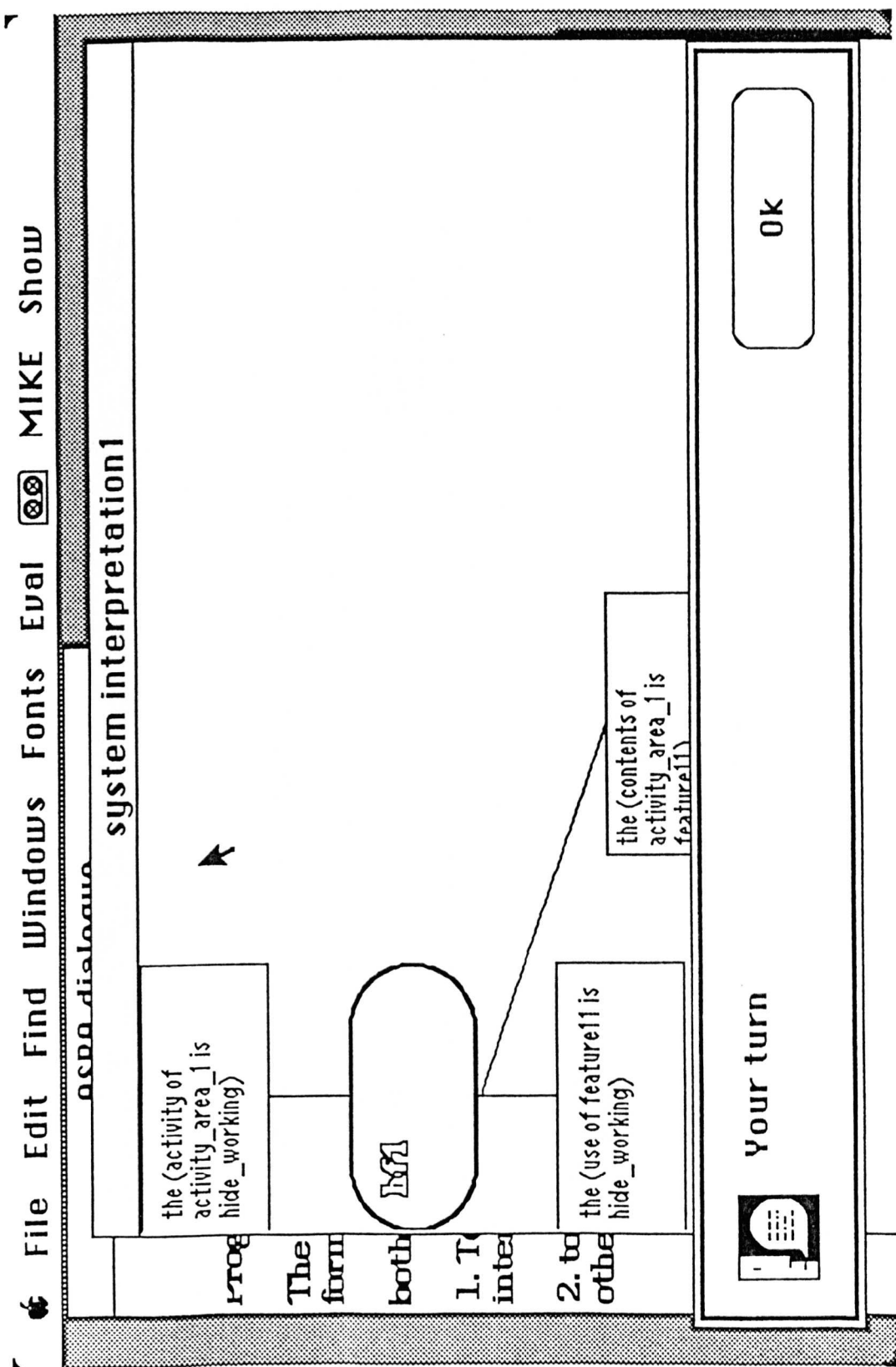


Figure 6.3

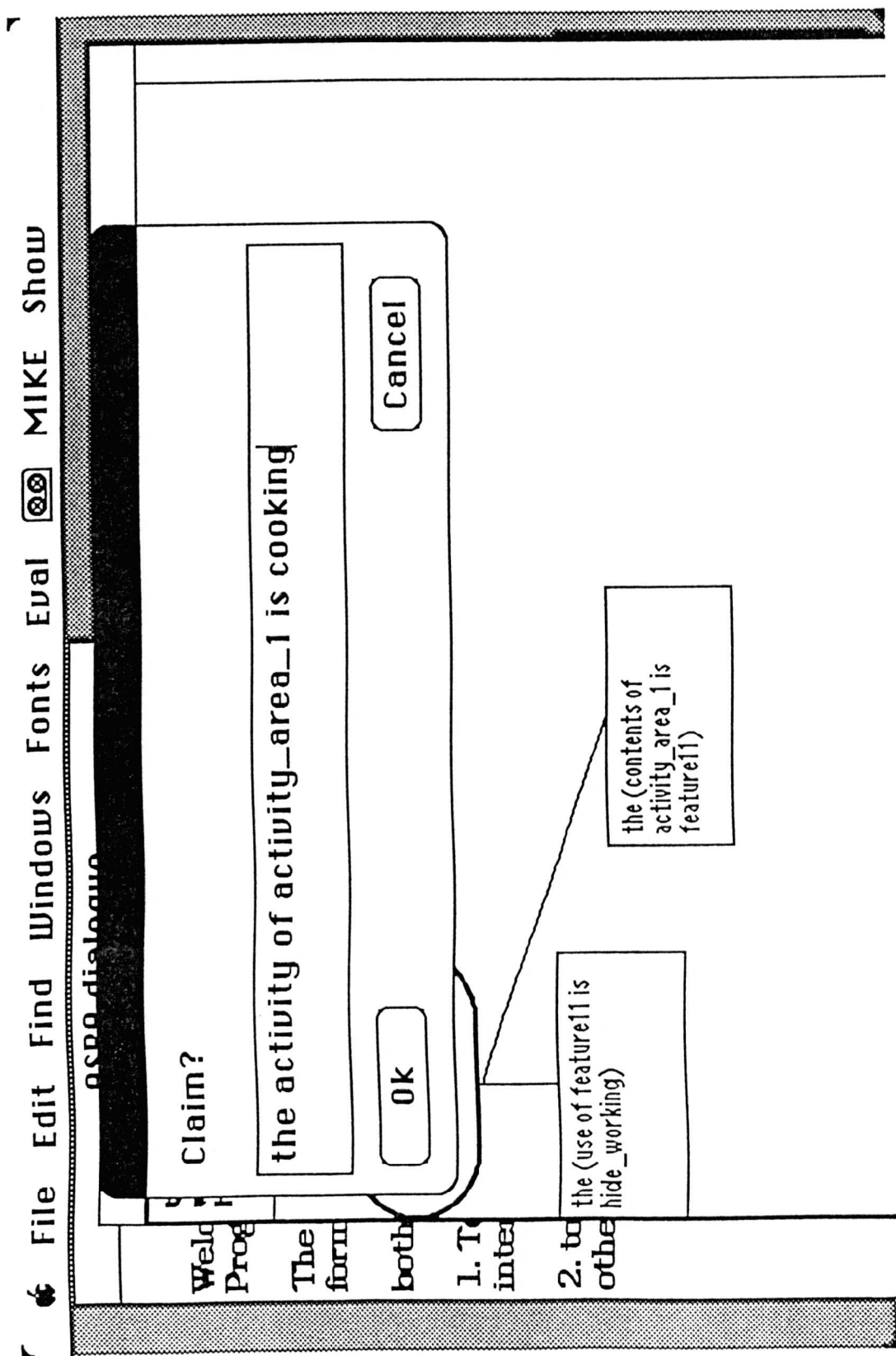


Figure 6.4

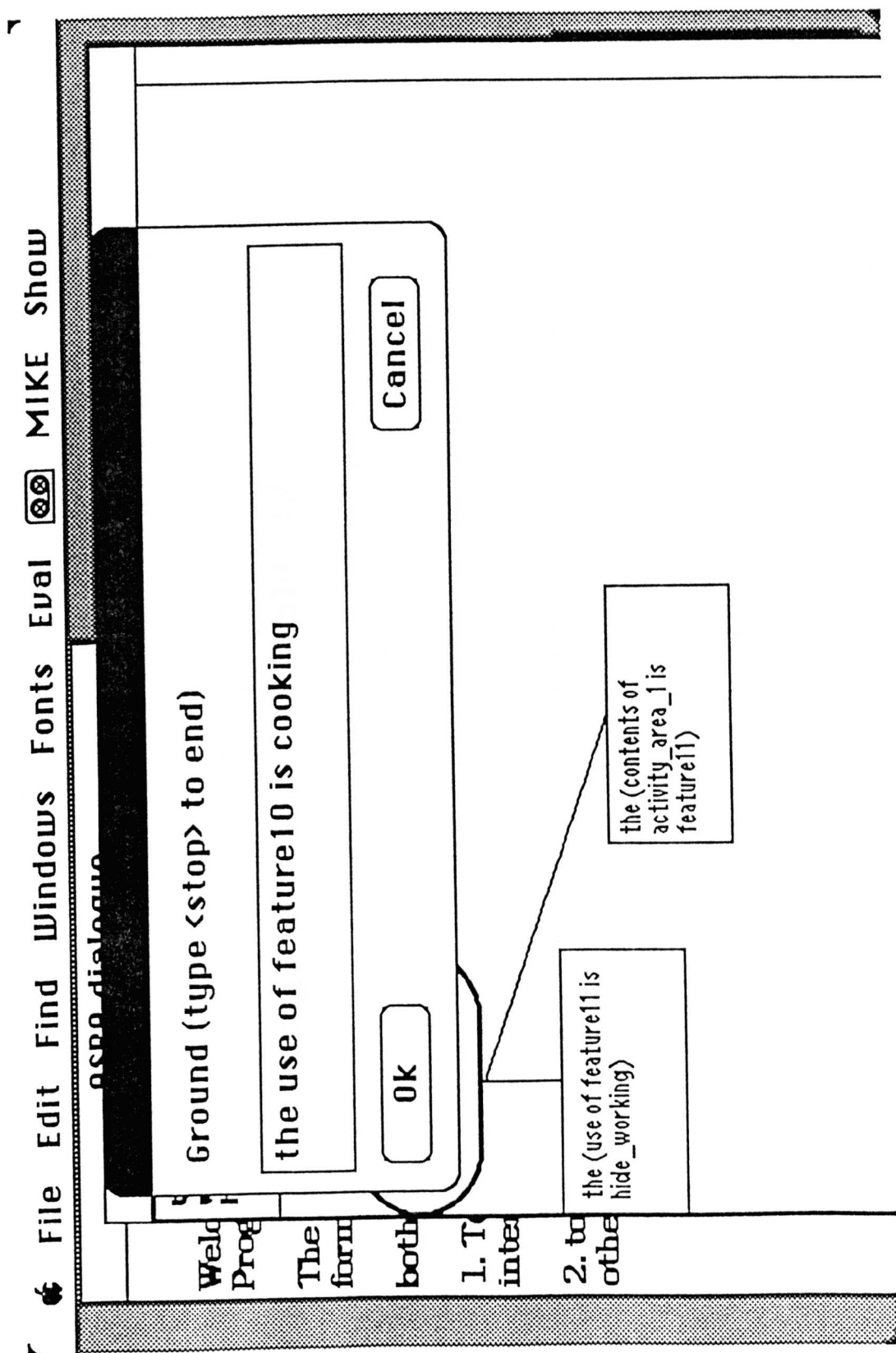


Figure 6.5

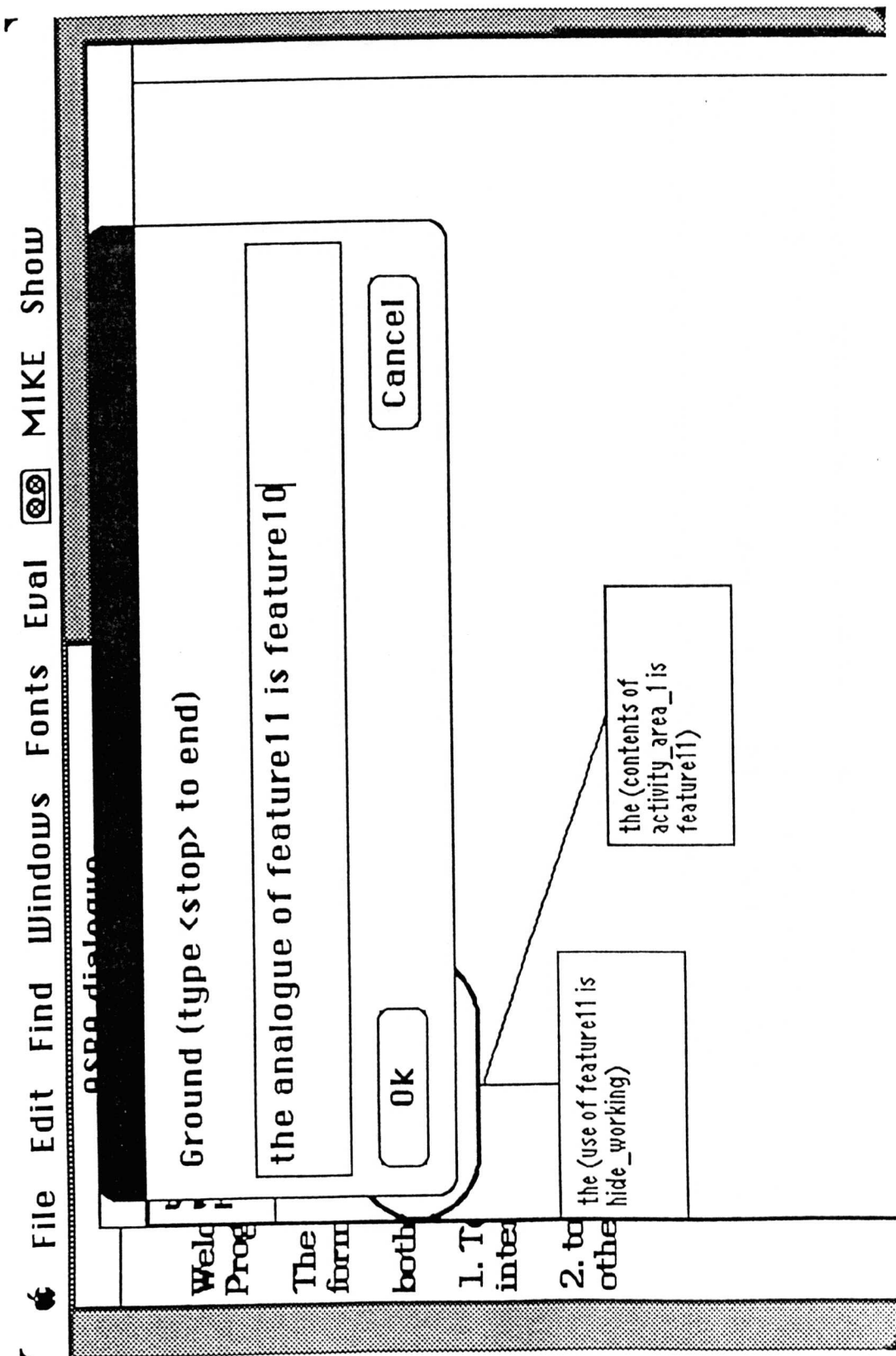


Figure 6.6

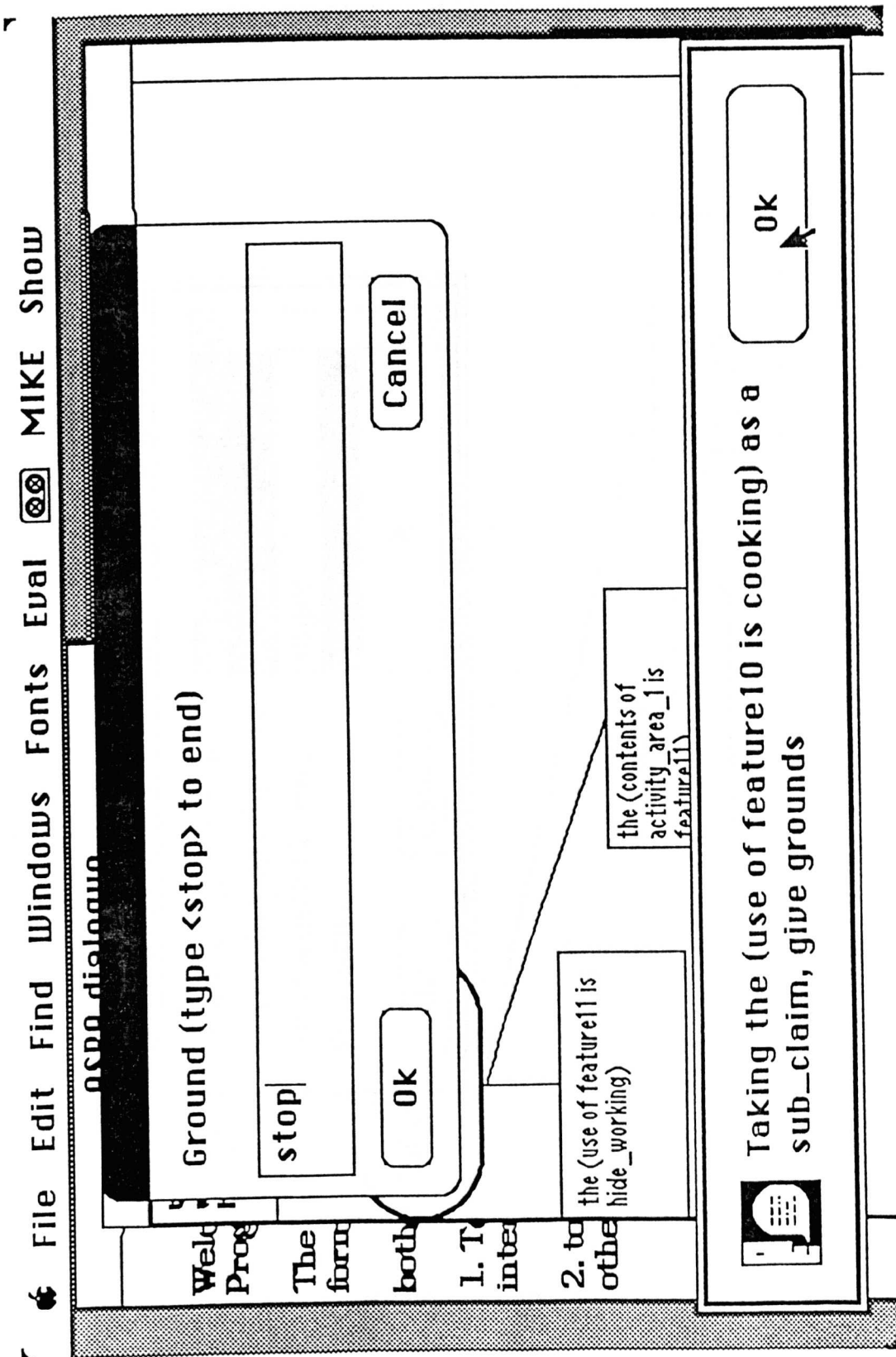


Figure 6.7

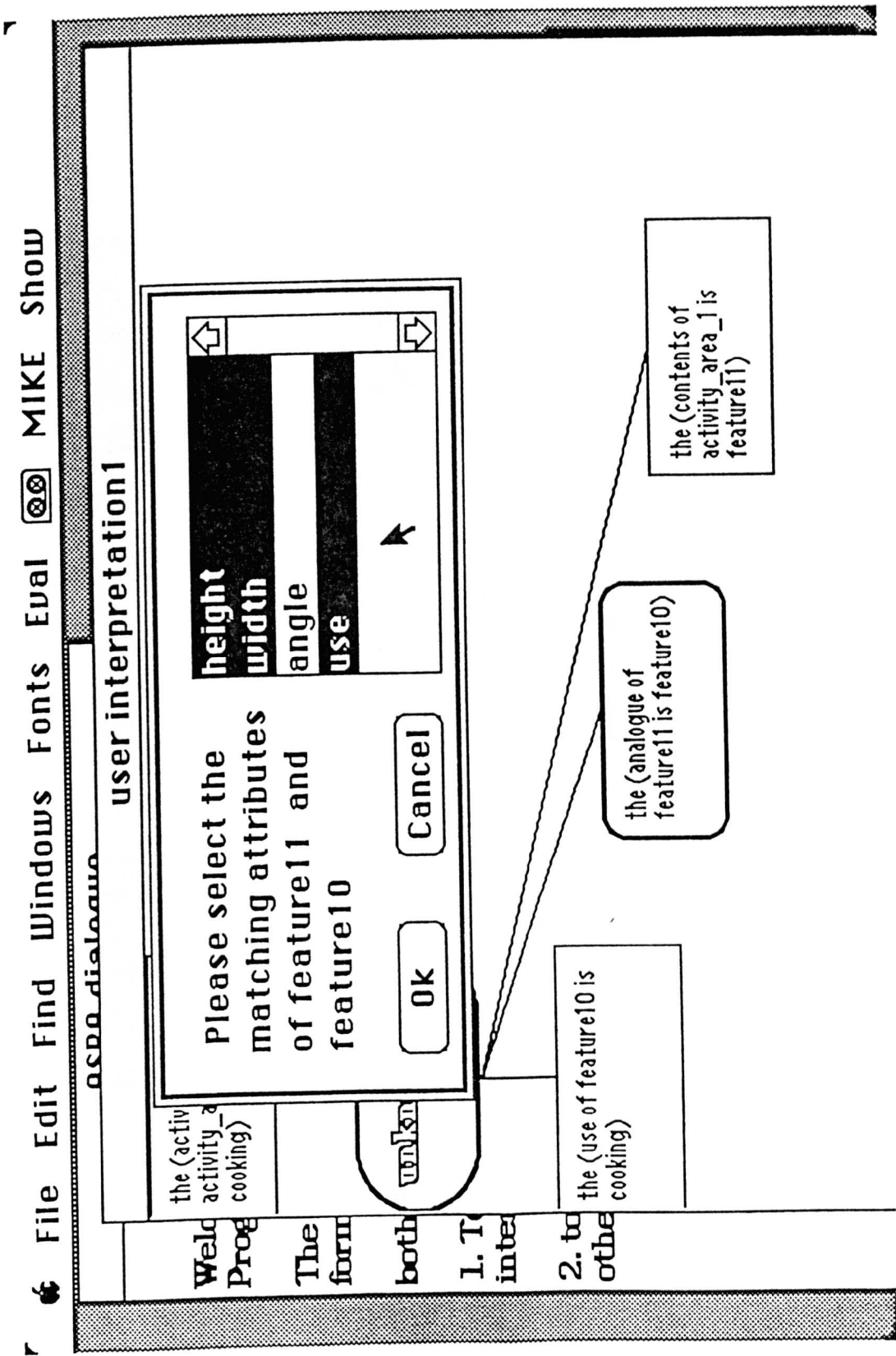


Figure 6.8

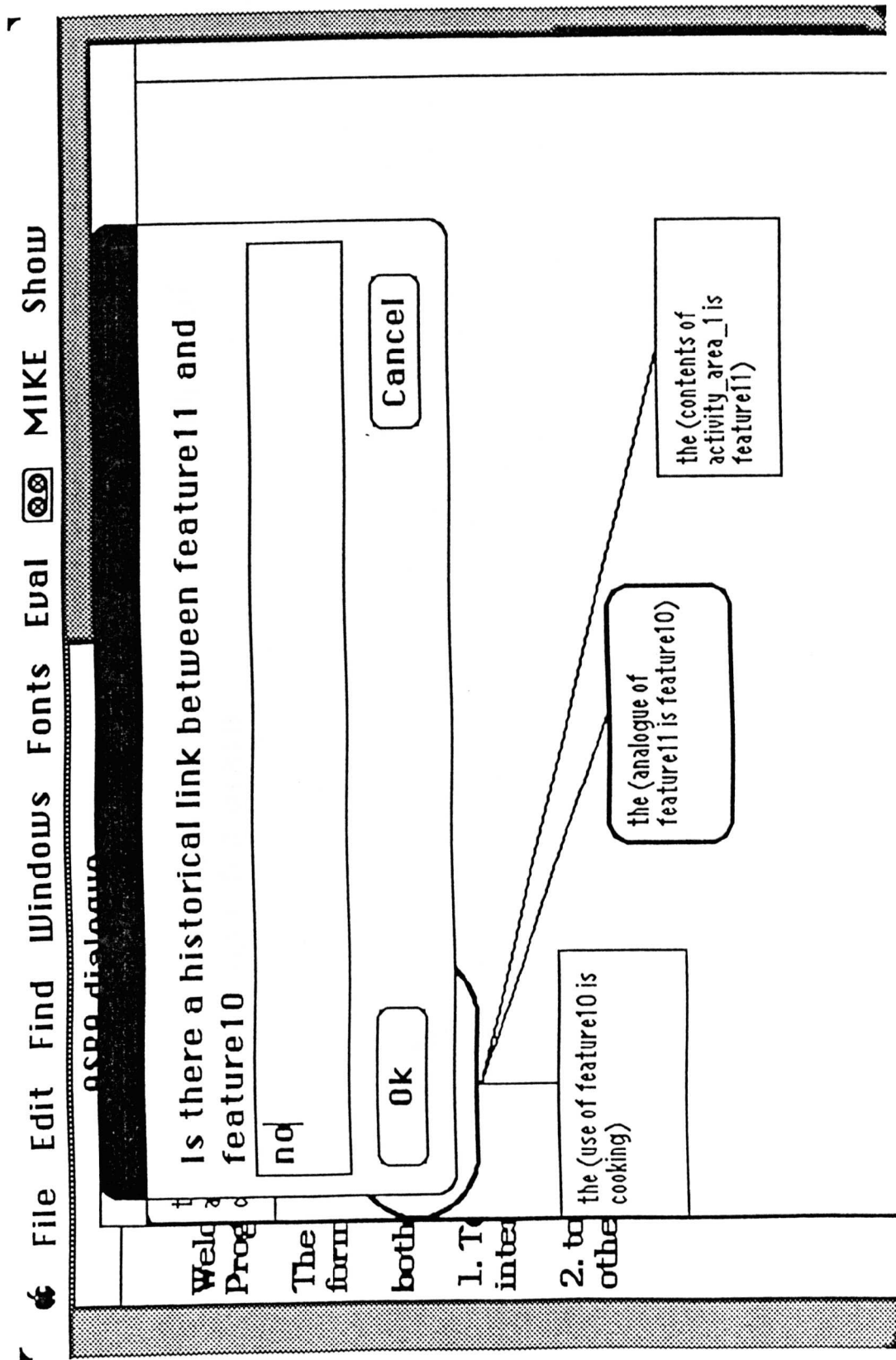


Figure 6.9

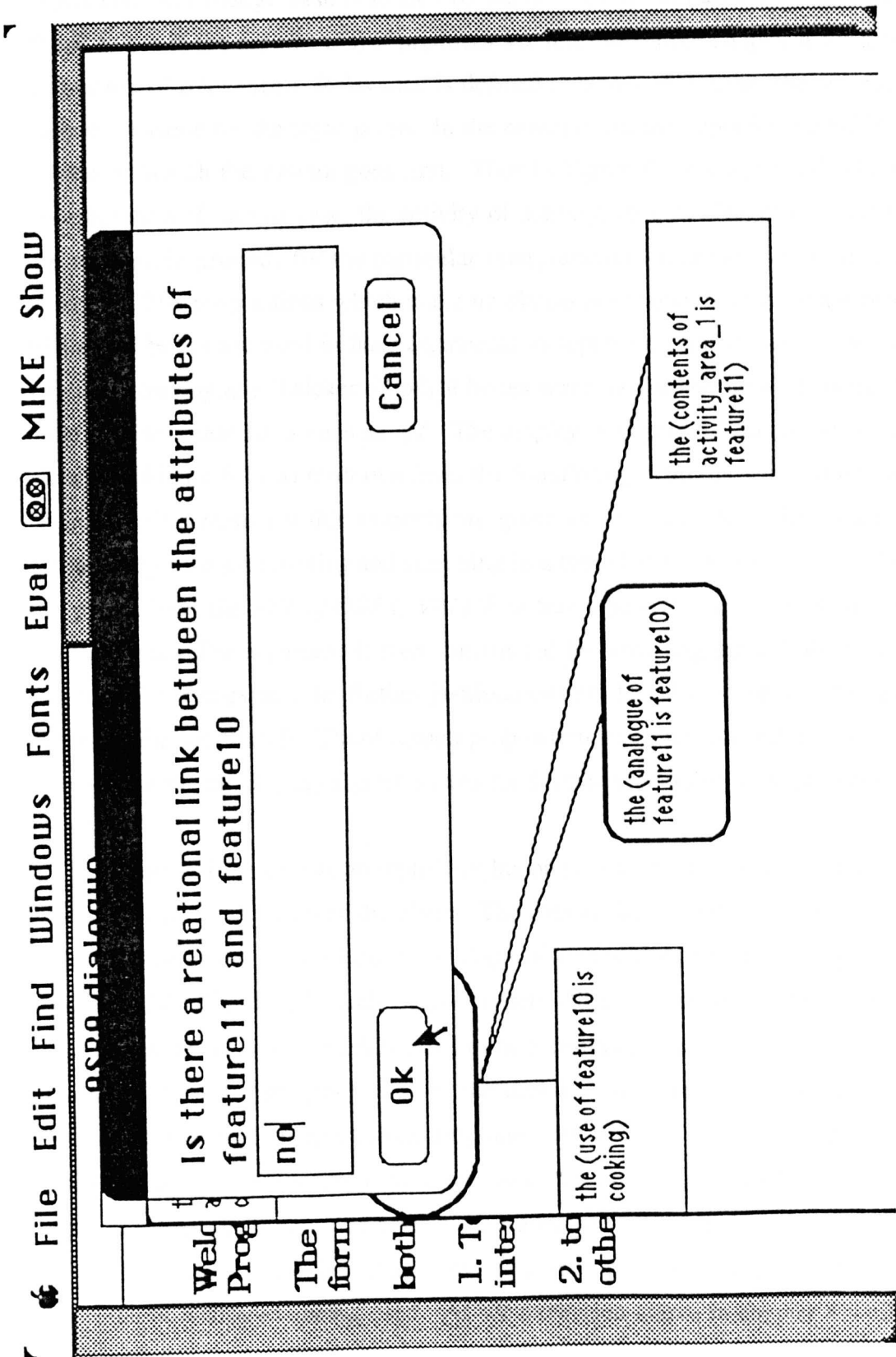


Figure 6.10

in everyday arguments; however, in the program, it functions as a means of narrowing the set of knowledge bases which have to be searched. In the examples the appropriate knowledge base is loaded by hand. The system also uses the topic as a check to see if the argument(s) it produces are relevant. (See chapter 4 for a fuller discussion of relevance.) Relevance is defined in terms of whether the argument supports a value for the topic given. In the example the user opts for the SUSU form of debate, in which the system goes first. Thus in Figure 6.3 the system displays an interpretation of, in this case, the activity of *activity_area_1*. The display shows all the supporting grounds for the particular interpretation - that the activity is hide-working. The propositions which make up claims are shown in rectangular boxes (Elliptical boxes are used in later arguments to represent grounds which mention the key word *analogue*.) Thicker elliptical boxes serve to display the warrant for the claim - in this case rules such as *hf1*. The display in Figure 6.3 can be seen more clearly in Figure 6.11(a) (redrawn from the MacProlog print out - the rather fuzzy Macintosh™ screens for this example are given as Appendix III). This argument is achieved by forward chaining and searching in a brute force way for a frame of which the proposition *the ATT of OBJ is VALUE* is true where *the ATT of OBJ* is the current topic. The argument is then constructed by searching through the justification facets of the frames until no further justifications are found (i.e. the ultimate grounds for the claim are found). These related propositions are then converted into a form appropriate for the display algorithm and for further processing as argument nodes and displayed.

In Figure 6.4 the user is prompted for his or her claim. In Figure 6.5 the user enters a ground in support of the claim. The user is then prompted for all the other grounds (Figure 6.6 shows how an analogy is asserted) and all the sub-grounds for these grounds (Figure 6.7) until 'stop' is typed. In this case we can take *feature11* to be a hide-working frame and *feature10* to be a cooking spit.

At this point the program displays the user argument in full (see Figure 6.11(b)) and attempts to find an appropriate response. In this case the system marks the node which contains the proposition *the analogue of feature11 is feature10* as of weak status: *analogy_ground*. The system then attempts to attack the weak spot it has found. Since the domain is archaeology the system deals with analogies by finding out what type of analogy is being used. As we have seen above in chapter 4 while

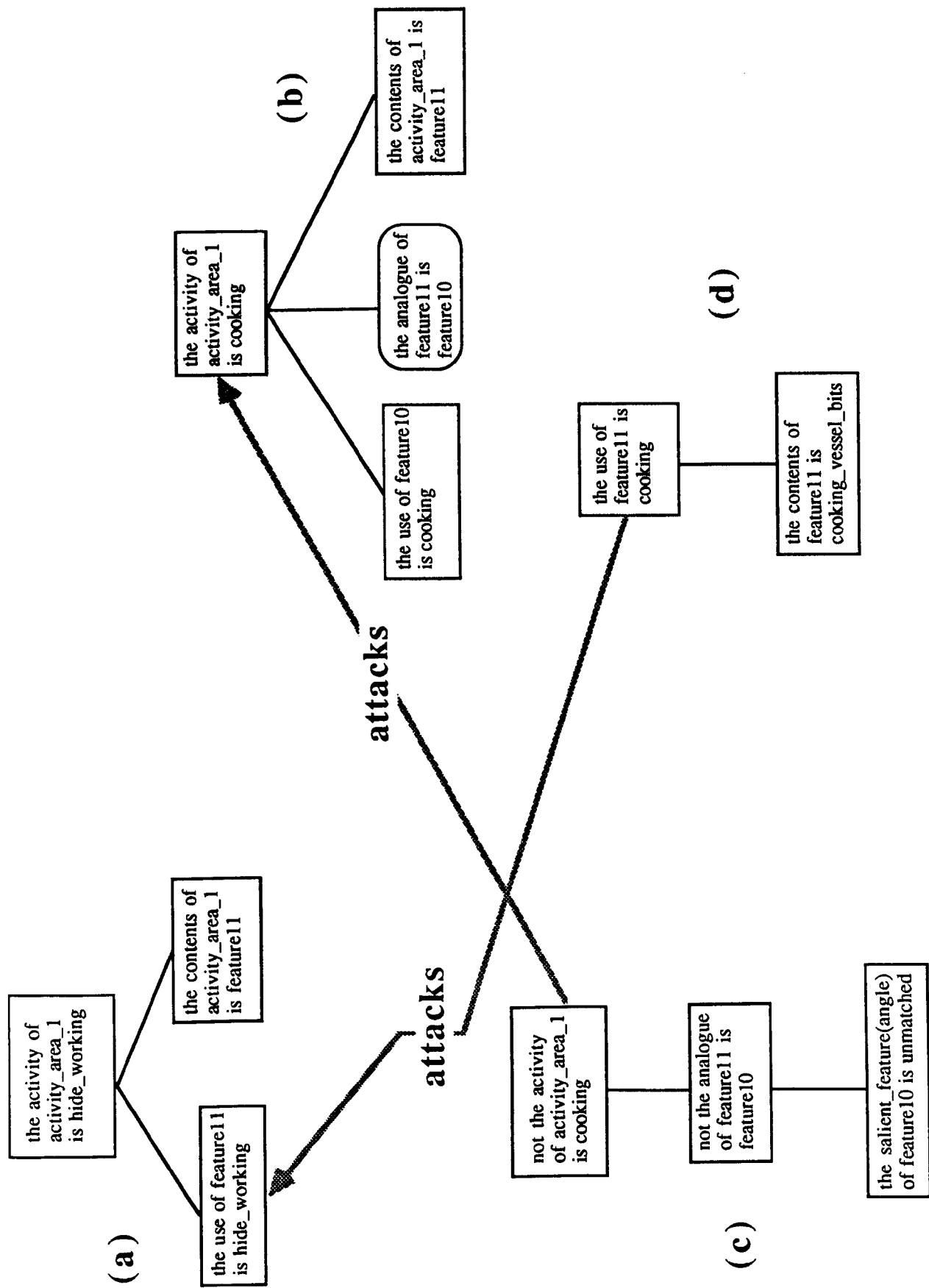


Figure 6.11 - Example 1(a) - redrawn screens

historical and relational analogies are acceptable in archaeology, simple analogies are not. If the field had been physics for example any analogy would be attacked.

In order to find the analogy type the system prompts in Figure 6.8 for the user to select the attributes which match between features 11 and 10. The attributes are taken from the frame (feature11) which is stored in the domain knowledge base as shown above. If the user selects all or a large number of the features then the analogy is simple unless there is a historical or other relation. In this case the user in Figure 6.8 suggests that the features have height, width and use in common (the significance of the omission of angle will become apparent in a moment). The system then goes on to ask whether there is a historical link between the features in Figure 6.9 and whether a relational link is held in common in Figure 6.10. In this example the answer to both questions is negative. Since the analogy is a simple analogy the system now attempts to deal with it in one of the two ways mentioned in chapter 4 above. In this case it finds that there is an attribute - *angle* - which has not been matched and which is salient for this field (or context). This is stored as a Prolog fact. The system constructs an argument of the form 'not User-claim' because 'not Analogy-ground' and displays this (Figure 6.11(c)).

The user is then prompted for his or her second argument. The user can elect to attack the system argument, defend own or alter own. In this case the user elects to attack the system argument. The user then enters an argument as before (Figure 6.11(d)). In this case the user's second argument is an attack on the system ground - *the use of feature11 is hide_working*. After displaying this, the system finishes its run.

The nodes which constitute the argument network (or argument-2) can be exemplified by the following:

```
sysarg0 instance_of argument with
  claim: (the activity of activity_area_1 is hide_working),
  grounds: [(the use of feature11 is hide_working),
            (the contents of activity_area_1 is feature11)],
  warrant: hf1,
  supports: [],
  supported_by: [sysarg2, sysarg1],
  external_supported_by: sysarg3,
  external_attacked_by: userarg3.
```

This frame represents the top level of the first system argument-1. It is internally supported (as part of the argument-1 whose main claim it contains) by *sysarg2* and *sysarg1* each of which has no further support. The *userarg3* slot value refers to the root node of the user's first argument-1, while *sysarg3* refers to the second system argument-1.

Example 1(b) - dealing with analogy by finding a contrary analogy

To avoid the prolixity of the above example I will deal more briefly with the remaining examples. In this example the system responds to the user argument by discovering a weak node: *analogy_ground* and then discovering an alternative analogy.

Here the user elects to present the first argument. Thus the form is USUS and in Figure 6.12(a) the user's first debating step is displayed. In response the system presents an alternative first argument (Figure 6.12(b)). There would be no point in presenting the same argument as that of the user. Of course, some elements of the two arguments might match. In this case the ground - *the contents of activity_area_1 is feature11* - is held in common. The user responds to this by attacking the system claim *the use of feature11 is hide_working* with an argument for the claim that feature11 is used for cooking (Figure 6.12(c)). The system responds by suggesting that the user's first argument fails because there is another analogue for feature11 - feature13 - which has a different use (Figure 6.12(d)). The point being that at the very least some doubt may be cast on the original claim. In this case the system uses its other rule for dealing with simple analogies to a) find another analogue and b) discover that it has a different use. Contrary analogue is thus defined in the implementation in terms of different *uses*. In a fuller version this might be done in terms of some more complex process of reasoning probably including some gap-filling.

Example2

The second example involves the use of viewpoints. This example depends on the provision of different viewpoints within the system (see chapter 5). There are two viewpoints - the system and the user. These model the differing sets of beliefs which are held by the system and the user. The system viewpoint includes the beliefs which the system is given or which it can infer. The user viewpoint represents the knowledge that the user either makes use of in arguments or which the system can

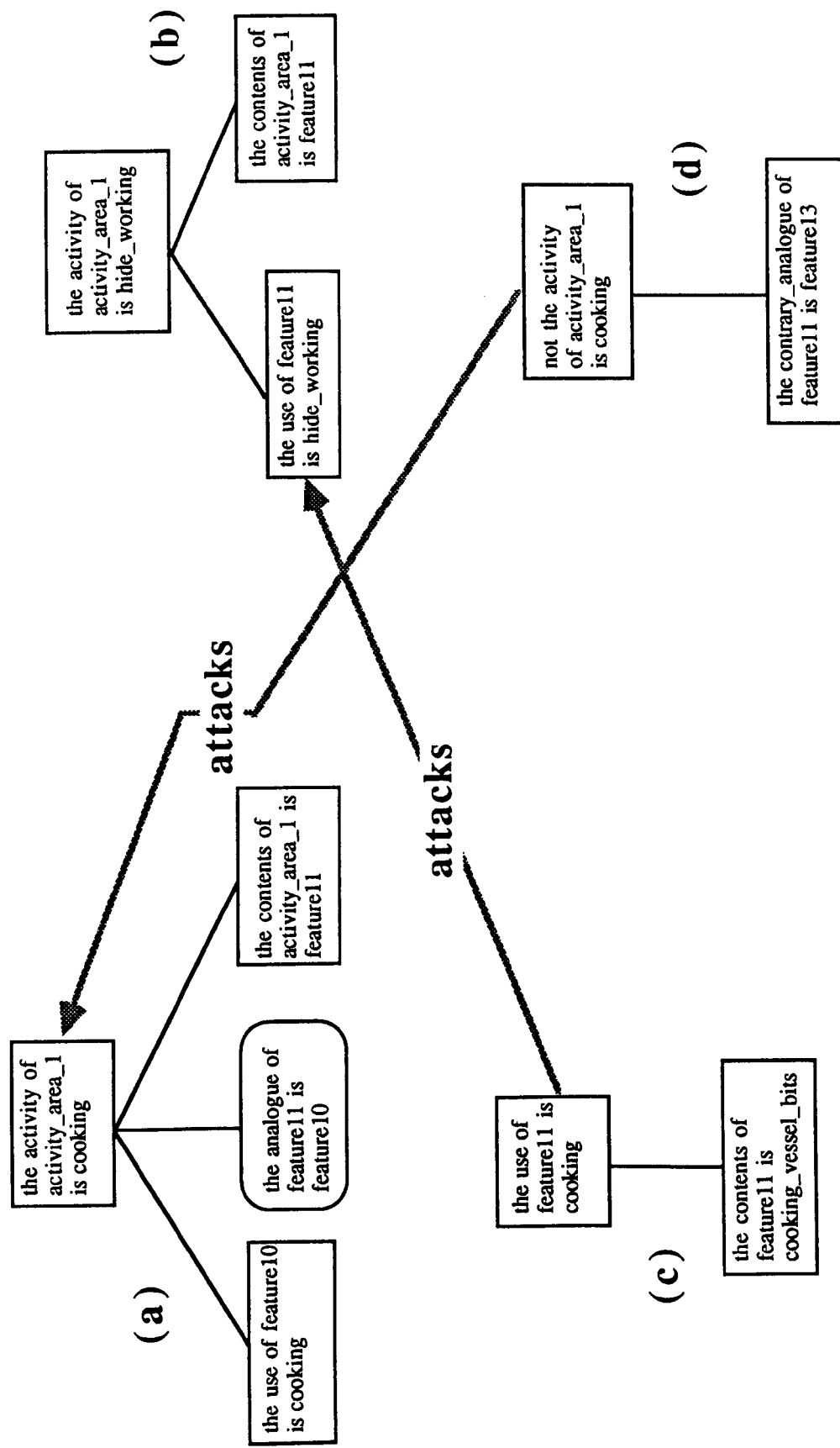


Figure 6.12 - Example 1(b) - redrawn screens

infer from this knowledge. The system can thus know about beliefs which it does not hold. This is implemented as separate partitions of the Prolog knowledge base. Since MacPROLOG™ requires that all of the same clauses to be in the same window the partitions are implemented using two clauses: *user_believes* and *system_believes*. These isolate the user and system viewpoints, respectively, to avoid unnecessary processing by the forward chainer.

The use of the viewpoint also allows the system to conduct what are arguments *ad hominem*. This has a slightly pejorative ring to it but it only represents an attempt on the system's part to impose some consistency on the user's arguments. Thus in the example the system finds a node which is weak because some aspect of the claim or its grounds depends on a belief which contradicts or is contrary to some belief already held in the user knowledge base. The system then reports this to the user in the form of an argument in which the claim is a denial of the user claim because the grounds include the use of a belief which is contrary to another belief held. In a full system the argument could proceed by refuting this e.g. showing that it is not contrary or by discarding one or other of the beliefs and so on.

The example used can be seen in Figure 6.13. The knowledge base is as follows and represents a fuller implementation of the knowledge given the Bonnicksen quote above:

```
%% FRAMES
```

```
feature11 instance_of structure with
    height: 2,
    width: 2,
    angle: 45,
    use: unknown.
```

```
feature12 instance_of structure with
    height: 2,
    width: 2,
    angle: 45,
    use: hide_working.
```

```
activity_area_1 instance_of activity_areas with
    centroid: [10,10],
    contents: feature11,
    discovery_at: deer_hair,
    activity: unknown.
```

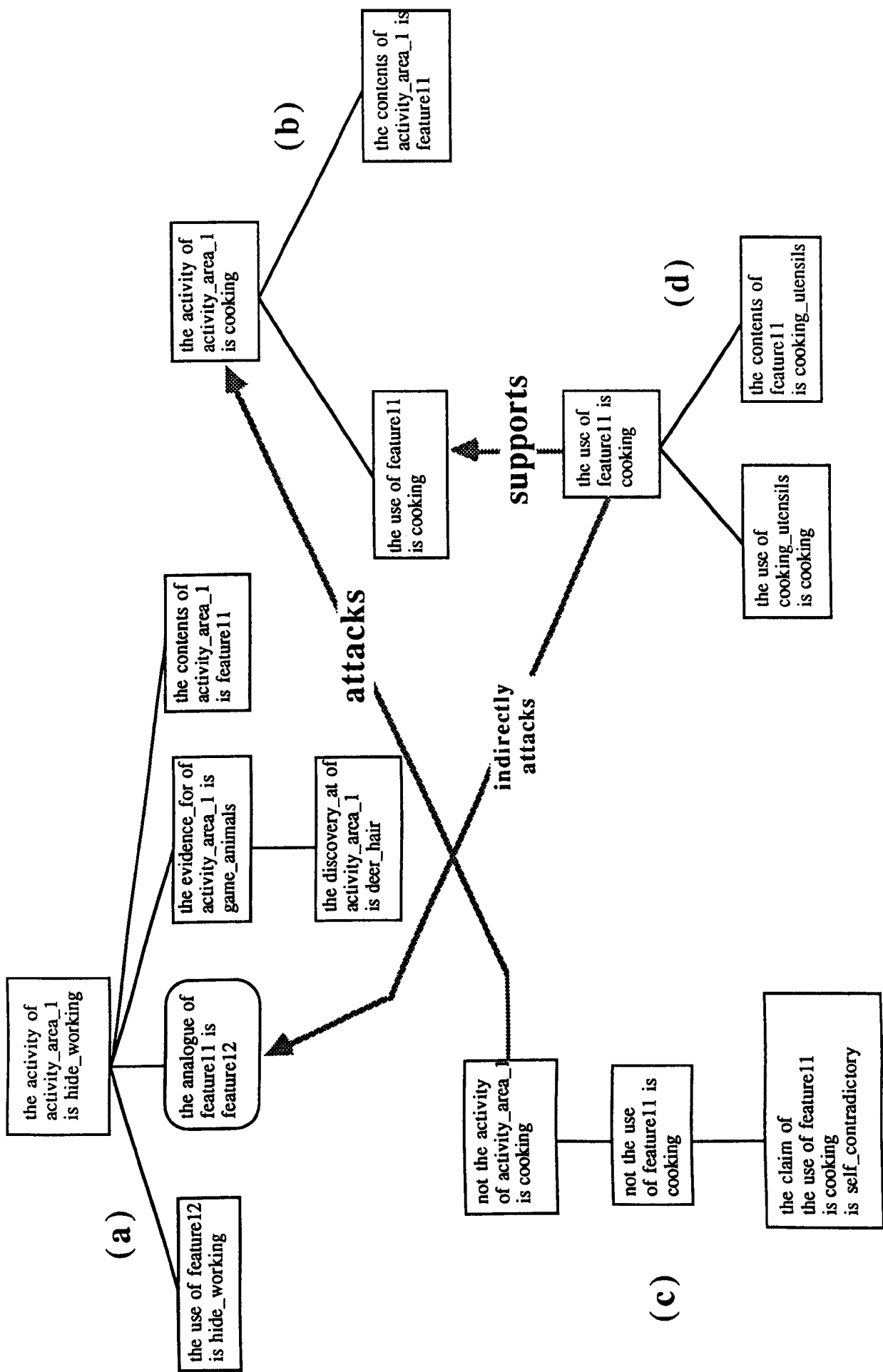


Figure 6.13 - Example 2 - redrawn screens

```

% RULES
rule hf1 forward
  if
    the contents of AA is S1 &
    the evidence_for of AA is game_animals &
    the analogue of S1 is S2 &
    the use of S2 is hide_working
  then
    the activity of AA is
      [value : hide_working,
       justification: hf1,
       grounds:
        [the contents of AA is S1,
         the evidence_for of AA is game_animals,
         the analogue of S1 is S2,
         the use of S2 is hide_working]].

rule hf2 forward
  if
    the discovery_at of AA is deer_hair
  then
    the evidence_for of AA is
      [value : game_animals,
       justification: hf2,
       grounds:
        [the discovery_at of AA is deer_hair]].

```

In this case the debate form is SUSU. The opening argument from the system is a simulation of the Bonnichsen reasoning and is shown in Figure 6.13(a). The opening user argument, however, depends on a claim about the use of feature11; that it is used for cooking(Figure 6.13(b)). In Figure 6.13(c) the system reports a self-contradiction which the user makes. In Figure 6.13(d) the user attempts to support this claim by providing an argument in support. It would then be up to the system to continue the argument in a full arguing system. The system finds the self-contradiction as before in two steps. In the first step the system searches through the user argument for self-contradictory nodes. According to its rules a node can be self-contradictory if the current node's proposition is contrary to some proposition stored in the user knowledge base or if the system can derive something from either which is contrary to

the other. All the real work is done here in this response so that in the second step the system has only to construct an argument based on the self-contradiction and display it.

Example3

This example depends on the storage by the system of theoretical principles which back up the warrants used in arguments. These, along with the warrants and facts, form the theoretical viewpoint of the system and, with the rest of the current argument-2, the context of the debate.

In this example the system responds to a user argument by attacking a principle which can be taken to ground one of the users warrants and hence indirectly attacks one of the user's claims. As in the above example the system searches each node for a weakness. In this case there is no self-contradiction or use of analogy. In accord with the precedence ordering given in section 5.3, analogy will be tried first since in archaeology it is the more common form of inference and must be identified as a possible focus of attack. Failing to find an analogy, the system successfully looks for weak principles. It must be noted, however, that, while this is an example of *level shift*, the argument style is not *level based* since no appeal is made to a model or set of principles as grounds in the argument.

The third example is slightly different from the others. It depends on the use of a different domain knowledge base and of an extended system knowledge base. For a fuller account of the archaeological argument surrounding the Wessex culture see Appendix II.

- knowledge base for example 3

```
%% FRAMES
```

```
sites_period_1 instance_of site_areas with
  contents: [crummy_pots,stone_tools,cremation_burials,beakers],
  period: neolithic.
```

```
sites_period_2 instance_of site_areas with
  contents: [wrist_guards,beakers,gold_earrings,
             gold_ornaments,bronze_daggers],
  period: eba.
```

```
laterthan(eba,neolithic).
```

```
% RULES
```

```
rule wessex_1 forward
```

```
  if
```

```
    the contents_nature of Sites_period_2 is new_artefacts
```

```
  then
```

```
    the cause_of_social_change of Sites_period_2 is
```

```
      [value : intrusion_of_outside_group,
```

```
        justification: wessex_1,
```

```
        grounds:
```

```
      [the contents_nature of Sites_period_2 is new_artefacts]].
```

```
rule wessex_2 forward
```

```
  if
```

```
    start &
```

```
    Sites_period_1 instance_of site_areas &
```

```
    Sites_period_2 instance_of site_areas &
```

```
    prolog(not Sites_period_1 = Sites_period_2) &
```

```
    all contents of Sites_period_1 are Contents1 &
```

```
    all contents of Sites_period_2 are Contents2 &
```

```
    the period of Sites_period_1 is P1 &
```

```
    the period of Sites_period_2 is P2 &
```

```
    prolog(laterthan(P2,P1)) &
```

```
    prolog(intersection(Contents1,Contents2,I)) &
```

```
    prolog(length(I,Len)) &
```

```
    prolog(Len < 4)
```

```
  then
```

```
    the contents_nature of Sites_period_2 is
```

```
      [value : new_artefacts,
```

```
        justification: wessex_2,
```

```
        grounds:
```

```
      [the contents of Sites_period_1 is Contents1,
```

```
        the contents of Sites_period_2 is Contents2,
```

```
        the intersection of contents is small]].
```

The example is shown in Figure 6.14. Again the debate form is SUSU. The field is as usual, archaeology, though the topic is now *the cause_of_social_change of sites_period_2*. In this example the system interpretation (Figure 6.14(a)) that *the cause_of_social_change of sites_period_2 is intrusion_of_outside_group* is grounded on a set of grounds and warrants which are concerned with the novelty of the grave--

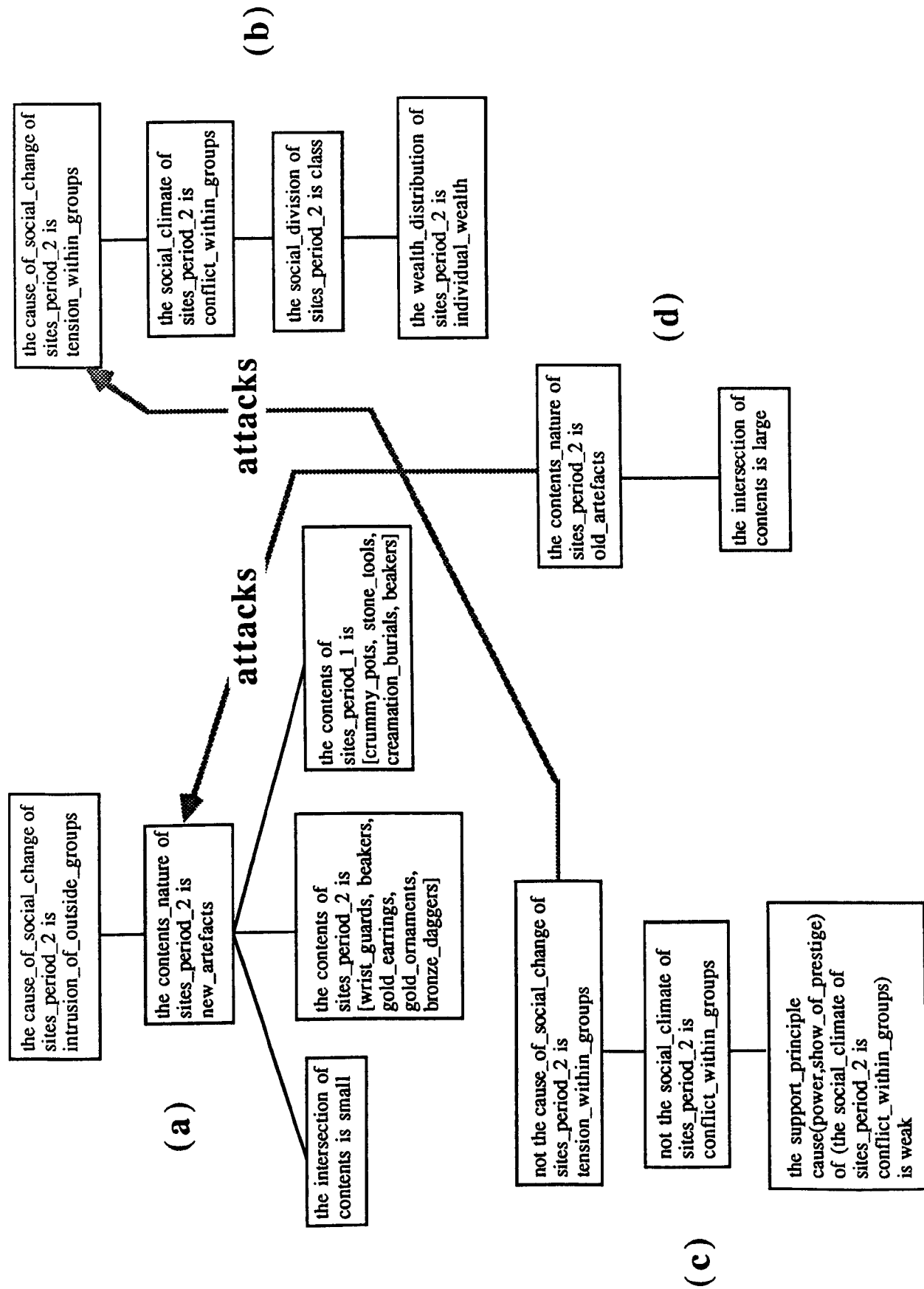


Figure 6.14 - Example 3 - redrawn screens

goods found in the burial tombs. The user response (Figure 6.14(b)) argues for the contrary claim that *the cause_of_social_change of sites_period_2 is tension_within_groups*. This is based on a chained argument about the causes of social divisions. In Figure 6.14(c) the system responds by attacking the user claim *the social_climate of sites_period_2 is conflict_within_groups*. In Figure 6.14(d) the user counterattacks by attacking one of the grounds of the original system argument. Again in this example we have a two stage process. In stage one the system searches the nodes for a weakness. Since none of its other rules succeed it attempts to find a node with a weak backing principle (see section 6.1.5 above). To do this it constructs a feasible warrant for the move from grounds to claim at that node and using this finds the appropriate backing principles. There is a single ground (G) at this node: *the social_division of sites_period_2 is class*. The claim (C) is *the social_climate of sites_period_2 is conflict_within_groups*. The system constructs the feasible warrant (*the social_division of sites_period_2 is class -> the social_climate of sites_period_2 is conflict_within_groups*). From this warrant the system can move from the backing to the set of principles stored as *marxism*. This move is stored as a Prolog fact:

```
backing_for((
    the social_division of sites_period_2 is class ->
    the social_climate of sites_period_2 is
        conflict_within_groups),
    marxism).
```

The following code selects the minimum valued principle:

```
min_valued_principle(Backing_set, OutPrinciple):-
    findall([Principle, Strength], findprinciple(Backing_set,
        Principle, Strength),Principlelist),
    minlist(Principlelist, [OutPrinciple, Min_strength]).

findprinciple(B,Pr,S):-
    principle(
        backing_set: B,
        name: Name,
        claim: Pr,
        grounds: Grounds,
        strength: S).
```

This process is illustrated in Figure 6.15. The system then searches through the principles which make up the `backing_set` until it finds the lowest valued principle. It returns weak status: *arguable_principle* and the particular principle. The weakest principle is determined by slots for strength which are included in the principles. These can be illustrated by the following:

```
principle(backing_set: marxism,
          name: marxist_principle_1,
          type: explanatory,
          claim: cause(class_divisions,
                       conflict_within_groups),
          grounds : common_sense_principle_2,
          strength: 7) .

principle(backing_set: marxism,
          name: marxist_principle_1,
          type: explanatory,
          claim: cause(power, show_of_prestige),
          grounds : common_sense_principle_3,
          strength: 5) .

principle(backing_set: historico_ecological,
          name: h_principle_1,
          type: explanatory,
          claim: cause(invasion,
                       conflict_within_groups),
          grounds : common_sense_principle_1,
          strength: 8) .
```

In the second stage the argument is again merely displayed by the system. The resulting argument seems fairly elliptical (perhaps a good human-like quality) since the principle *cause(power, show_of_prestige)* does not seem to be immediately related to the claim *the social_climate of sites_period_2 is conflict_within_groups*. However it must firstly be remembered that it is the inference from *the social_division of sites_period_2 is class* which is being attacked. It is relatively easy to construct a chain of argument between the principle and the claim. It must also be remembered that the system makes no claim to including a fully worked out

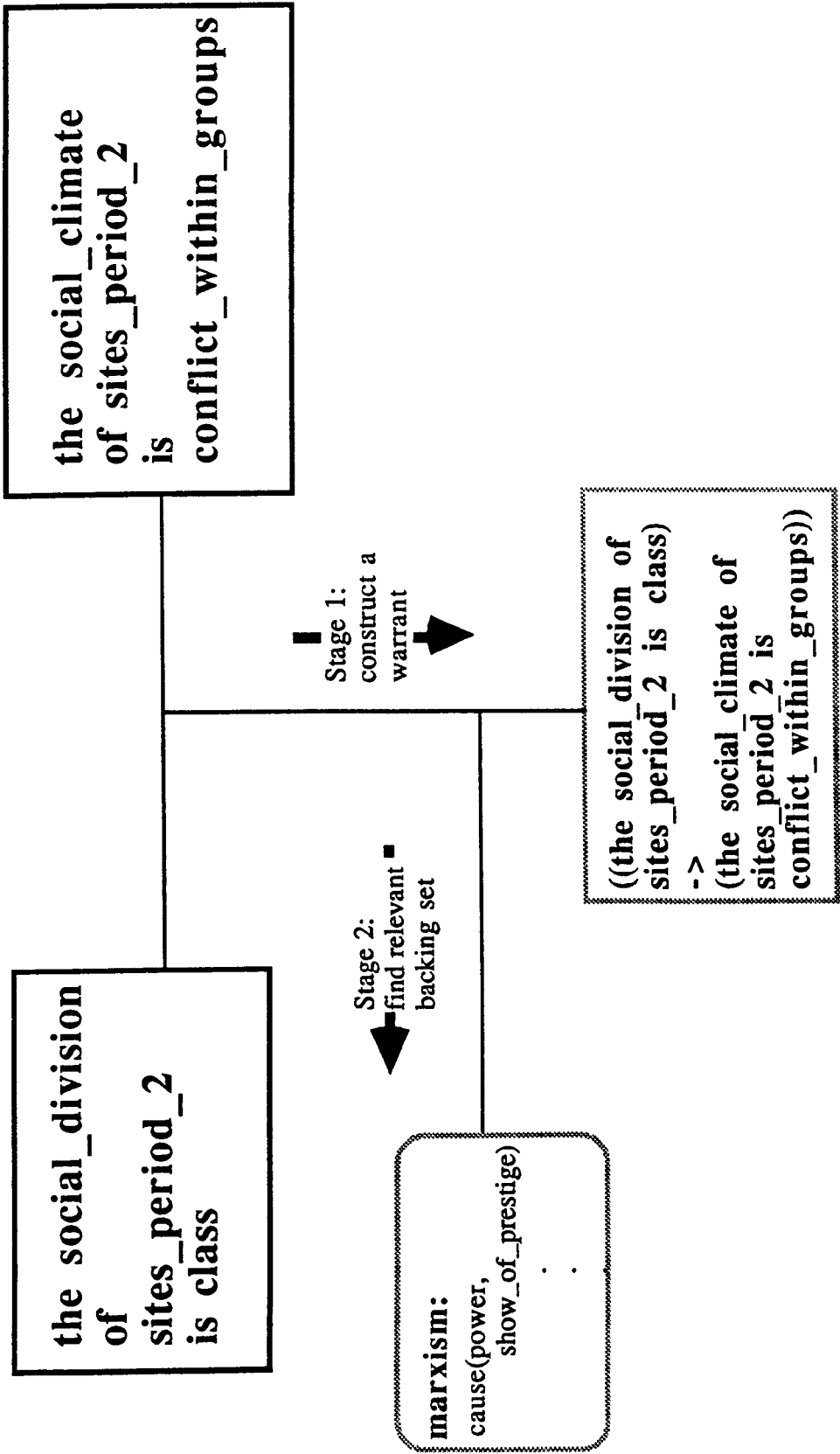


Figure 6.15 - Finding a weak principle

marxist theory. It is only meant to be suggestive of how an argument based on marxist principles might be attacked. Anyway many human arguments proceed in just this oblique manner.

Before turning to the extensions to the prototype which have been made apparent by experiments with the above examples it is worth summarizing for the reader what exactly the system requires in order to be able to function as it does. In general the system needs: (a) a suitable knowledge base and (b) code which can identify and deal with the particular argument type used. The knowledge bases and code have been presented above. In addition to these the system requires various values which in the implementation have been set up to allow the examples to work. Thus in example 1(a) the system simply stores the fact that the angle is salient. In 1(a) and 1(b) the system relies on the user inputting the attributes of the object and asserting that it is not historical or relational. In example 2 the system and user knowledge bases are set up for the example. In example 3 the strength of the principles is set up as is the particular 'constructed' warrant.

6.3 Future extensions to ASParch

The prototype program described above would need several additions and emendations in order to fully embody the model from chapter 4 or to function as a usable tool for archaeologists or others. The principal areas in which changes or new modules are needed are: the reasoner; the graphical interface; the weakstatus rules; assessment; knowledge base refinement.

6.3.1 The reasoner

There are four main ways in which the reasoner must be extended.

- The procedure *can_derive* must be fully implemented so that both forward and backward chaining are possible and so that gap filling is possible.
- Extensions must be made either to the interpreter or to the knowledge base to allow causal reasoning and perhaps other forms of plausible reasoning. These extensions would deal with what I have called *mode based* argument style in chapter 4. Further extensions need to be made so that the system can cope with arguments which appeal directly to models or principles (as opposed to making use of warrants which are grounded in backing theories). Appeals to models would entail the use of tokens of

mode based reasoning types to operate on values derived from models whether these are represented propositionally or as some form of simulation. Appeals to principles would require the use of transformative operators on representations of these principles.

- The analogy rules need to be made more sophisticated. For instance, to take account of the ideas of Hodder and Kedar-Cabelli discussed in chapter 4. If they are right, what counts as a good analogy depends on the theory held about the relations between the attributes of the possible analogues. Finding an analogy is not simply looking for matching attributes or second order relations. It involves finding a theory which relates the attributes appropriately (Hodder) or in the light of which the objects can be seen to have the appropriate attributes (Kedar-Cabelli).
- More use could be made of the warrants. At present the user is not asked for a warrant. This could be done and the warrant proffered could be used in checking the user's argument.

6.3.2 The graphical interface

There are two main areas where extensions need to be made.

- It should be possible to click on any argument-0 which makes up an argument-1 and get a display which includes as much information as possible about the warrants and backing for the move from a set of grounds to a claim. It should also be possible to move through the argument-1 by moving from one to another of the argument-0s which constitute the argument-1. This sort of facility is perhaps more easily provided in a hypertext environment and has not for that reason been attempted here.
- The implementation displays argument-1s or interpretations and the argument-0s (or graphical toulmin structures) but is not capable of displaying the full debate or argument-2 structure. To some extent this latter is not needed in a debate where the forms are fairly stylized but it would be needed in a full-scale argument. All the information needed to provide this display is stored so that the real difficulties that remain here are not artificial intelligence ones but simply to do with getting an algorithm which displays the largish trees in a comprehensible and tidy manner. With the addition of the graphical argument-network, the resulting set of displays would constitute a hypertext like document in which links between parts of the argument can be traversed easily to show different levels of detail. In other words the user can have visual access to either the argument-2, its component argument-1s or their

component argument-0s. Ideally the user would be able to select parts of the display and zoom in on these (see similar work on Prolog debuggers by my colleagues, Eisenstadt and Brayshaw, 1987). The user should also be able to select the proposition or propositions for counter-argument by interacting with the display. It might also be possible to use colour so that the different contexts and sub-contexts for reasoning which make up an argument-1 could be displayed in an exciting and informative manner. Again it seems to be outside the scope of this thesis to deal with what are largely problems in computer graphics and which have largely been solved already. In chapter 8 I will discuss a possible way of extending ASParch within a hypertext environment.

6.3.3 The weakstatus rules

I have only tested fully the weakstatus rules mentioned above. However I have implemented but not fully tested other weakstatus rules. This section presents details of these. Some are extensions to rules mentioned above while others are new.

- selfcontradiction

The following rules show how code could be invoked to see if a contradiction can be derived by making inferences from either P or X or both. The implementation of `can_forward_derive` would involve the invocation of the forward chainer and the subsequent check for appropriate propositions.

```
weakstatus (P,G,W,selfcontradictory(X)) :-
    user_belief(X),
    can_forward_derive (X1,X),
    contrary (X1,P) .
```

```
weakstatus (P,G,W,selfcontradictory(X)) :-
    user_belief(X),
    can_forward_derive (X2,P),
    contrary (X,X2) .
```

```
weakstatus (P,G,W,selfcontradictory(X)) :-
    user_belief(X),
    can_forward_derive (X1,X),
    can_forward_derive (X2,P),
    contrary (X1,X2) .
```

- OK status

The system would return OK if it can in fact derive the claim P from the ground G using the warrant W. The predicate would include a variety of inference methods. It could try forward and backward chaining in an attempt to find a connection between P and G.

```
%% ok status
weakstatus(P,G,W,ok):-
    can_derive(P,G,W).
```

The converse is given by the following:

```
%% cannot make inference at all
weakstatus(P,G,W,false_inference):-
    not can_derive(P,G,W).
```

- Unknown fact

One main way for an argument to fail is if faulty evidence is used. In this case the system can only tell that the fact is not known to it. It may find it in the user knowledge base or it may need to query the user for confirmation of the fact.

```
%% an unknown fact
weakstatus(P,G,W,fact):-
    on(F,G),
    not prove(F).
```

- Too few grounds

This represent the case (see troublemakers *external factor* and *neglected critical distinction*) where the system can derive the claim P from G only if other grounds are added.

```
%% not enough grounds
weakstatus(P,G,W,too_few_grounds):-
    can_derive(P,G1,W),
    sublist(G,G1).
```

- Wrong warrant

The system can derive the claim from the grounds using another warrant.

```
%% wrong warrant
weakstatus(P,G,W,wrong_warrant(W1)):-
    not can_derive(P,G,W),
```

```
can_derive(P,G,W1) .
```

- The use of causal reasoning

This could be marked (as in the case of analogy) with a keyword *cause*. On the other hand the system could make some inferences from the content of the proposition in question to its type as causal. In either case causal reasoning will be weaker than the deductive reasoning represented (if somewhat inadequately) by the rule-based methods.

```
%% causal model
weakstatus(P,G,W,causal_model):-
    on(G1,G),
    G1 = (the cause of X is Y) .
```

- Gaps

Gaps may be found which can be filled

```
%% gappy
weakstatus(P,G,W,gap):-
    not can_derive(P,G,W),
    can_fill_gap(P,G) .
```

or not as the case may be

```
%% unfilled-gap
weakstatus(P,G,W,unfilled_gap):-
    not can_derive(P,G,W),
    not can_fill_gap(P,G) .
```

6.3.4 Assessment

While I have suggested one way of implementing a form of assessment using numerical values in section 6.1.7, I am unhappy about this for a variety of reasons. These have already been rehearsed in my discussion of assessment in chapter 5. My main problem is that the rationale for assessment is not available for argument. A secondary worry is that the method I have adopted for weighing the strength of nodes - the ratio of links to weak links - does not take account of the fact that some parts of an argument are more *telling* than others. A weakness in what should be a telling argument is important while weaknesses in other parts of an argument with a telling node are irrelevant. Finally the assessment should take account of, say, the different types of analogy. Assessment is thus to some extent domain-dependent. This

implies that the weakstatus rules should return some weighting for the weakness of the node. For instance, selfcontradiction should incur a greater penalty than the use of simple analogy.

Thus I would like to see extensions of three kinds in an attempt to deal with assessment while permitting the result to be arguable.

- It would be possible to extend the expectation based reasoning used in KIVA, the archaeological interpreter implemented by myself and a colleague (Patel and Stutt, 1988, 1989). This technique is used to cut down on the number of interpretations. Here it could be used to determine which interpretation is best in relation to a model of an ideal site. However this could meet the objection (raised by Gardin in a personal communication) that the use of an ideal model does not by definition provide a means of dealing with previously unknown knowledge which might figure in what constitutes the best interpretation. The use of the ideal model would entail that this new knowledge was rejected.
- The meta-level assessment discussed in chapter 5 could be implemented. This would allow the system to make an assessment which was subsequently arguable. Argument would proceed at the level of backing for the assessment rules. However as I have said already, there are great difficulties here providing a viable set of meta-assessment rules.
- The system could be extended to cope fully with the multi-dimensional nature of arguments. Thus every proposition including warrants and backing principles and assessment results could be arguable.

6.3.5 Knowledge base refinement

There are two aspects to this: adjustments to the knowledge base as the result of argument exchanges and the acquisition of new knowledge.

- The relationship between assessment and knowledge base alteration is close. For it is as a result of a negative assessment that the system will need to alter its knowledge base. Assessment also depends on the values which are stored in the knowledge base. If numerical values are used, a simple means of implementing the outcome of the assessment process is possible: That is, changes in the system or user knowledge base could involve strengths of belief stored as slots. When the system loses an argument the appropriate strengths are lowered and conversely if the system wins. It is psychologically implausible to suggest that a belief will be given up

altogether unless an overwhelmingly convincing rebuttal is given. It is therefore more likely that a belief will fall out of use over a period of time. This can be rendered by successive diminishment of the values for strength of belief over a series of arguments. This has not been implemented for the prototype program since there seems little point in implementing this for a system which is only capable of one formal debate exchange. (For what it's worth an untested piece of code for evaluation of strength of belief is given as the last page of Appendix I.) As I have already suggested, I am unhappy with the use of numerical values. They themselves are psychologically improbable. It would be possible to use some symbolic terms to represent different values. This is more plausible and indeed reflects the intuitions of archaeologists. Again however it suffers from similar defects to the numerical approach. The best approach seems to be that of researchers in Truth or Reason Maintenance Systems (Doyle, 1979, 1980) and indeed the ASP including assessment and knowledge base refinement acts as a sort of non-monotonic reasoner.

- As the process of arguing goes on, the system viewpoint will come to include elements of the user viewpoint if these are supported sufficiently. On the other hand the system viewpoint may by the same process lose some of its contents. Of course, the user viewpoint is always provisional since the system may be erroneous in its ascription of beliefs. By this means the system acquires new knowledge about facts and warrants and how these are related to other elements in its knowledge base. This includes knowledge of the attributes of objects and the analogies between objects. An ASP does not only produce a set of unrelated rules and facts but a structured high level representation of the relations between these in the argument-1s and argument-2s.

6.3.6 Response

In general a more complete model of debate would allow multiple argument-1s per move. Thus response would have to take account of this and attack multiple nodes. In turn this would result in a more complex representation for the argument network and more work for the assessment module. The system should be able to gauge whether or not an opponent has made an implied rather than explicit attack on some part of its argument and hence whether or not that component needs support. An entirely new situation may be brought about by the user's first move (that is, the context changes). Thus the system looks for weak spots in its own argument and

attempts to salvage these before attacking the weak spots in the user's argument.

In dealing with analogies the system should be able to deal with historical analogies (perhaps by attacking the historical links) and relational analogy (perhaps, in the light of the discussion of analogy in section 6.3.1, by attacking the theory which grounds the relation).

In generating arguments the system should go beyond the mere statement that a weak spot has been found to a full counterattack.

6.4 Conclusion

In this chapter I have presented a partial implementation of the program design discussed in chapter 5. While the program has only been fully tested on the examples given, ASParch could perform similarly for any topic within the domain of archaeology if the appropriate knowledge base were available. By appropriate, I mean one which has facts stored as MIKE frames and rules coded using the MIKE syntax. Indeed the system as it stands could deal with arguments from any domain if there is a knowledge base encoded in a suitable fashion. In this case, however, domain specific use of analogical arguments would not be used. Argument claims and grounds must also be framed in terms of the syntax available in MIKE as in the above examples. With the extensions proposed in section 6.1.3 ASParch would approximate to a complete implementation of the program design of chapter 5. In the next chapter we will investigate the possibility of applying the ASP approach to other domains.

-o0o-

Chapter 7 - Other argument fields

Argument: The ASP model can, in theory, be used in any field or domain which is similar in appropriate ways to archaeology. Such domains include: literary criticism, legal argument and darwinian theorizing. Hypothetical examples are presented to illustrate how an ASP might work in each of these domains.

In developing the design for an ASP I have concentrated on applying the model to the domain of archaeology. However, the model can be applied to other domains. In section 7.1 I discuss the possible use of an argument support program in literary criticism; in 7.2 the application to legal argument; and in section 7.3 to Darwinian theory. I would thus claim that an ASP can be widely applied both within and without the humanities.

Before turning to a consideration of the use of the ASP model in literary criticism I will outline the criteria for a successful application of the model to a domain. These are derived from a consideration of the sorts of features which are required in designing a system to support archaeological argument.

- Possible domains must be interpretative, justificatory or explanatory disciplines since it is the differences between argument-1s embodying these which form the basis for argument-2s. The model given above deals primarily with argument-1s as interpretations. I have also suggested that both explanations and justifications can be regarded (in some cases at least) as sets of propositions in support of a conclusion or claim. In general argument-1s will be generated in any field where the data is uncertain, incomplete or changing. While not all such fields can be labelled as interpretive many will include an element of interpretation. Taken in this way a discipline is interpretive (or includes interpretive elements) if there is no certain means to derive results. If there were it would be calculative and there could be no argument.

Interpretation is of course a widespread process. We do it every day in decoding the signs and symbols which surround us. We also interpret when we hear utterances and see images. However for a debate to arise about these interpretations and hence

for interpretation to be an argument generator, the interpretation must be at a conscious level. Thus while it may be possible for systems such as Hearsay (Erman et al., 1980) to interpret utterances, there is no role here for debate. Indeed it is psychologically improbable that at that level debate takes place. Of course as the interpretation rises higher in the semantic hierarchy debate becomes possible. The dividing line is unclear. However, in all the examples discussed here - archaeology, literary criticism, the law and darwinian theory - the interpretation (or argument-1) and hence the debate are at a conscious level.

The following represent criteria for the *optimum* use of an ASP. However, while they spell out the characteristics of a domain which an ASP is designed to deal with there is no reason why an ASP should not deal with a domain which, for instance, uses only deductive reasoning and never appeals to theory. It's just that there seems little point in this since there would be little room for argument about the decisions of such a system.

- Possible domains can use analogical reasoning and other forms of plausible reasoning. An ASP can deal with analogy as one form of non-deductive reasoning.
- Possible domains can make use of theories internal to the domain in their interpretations. (Internal and external theories will be illustrated in the next section). For the system to build the appropriate Toulmin structures this criterion is necessary. The level shift exemplified in example 3 from Chapter 6 above shows how the system can make use of this knowledge.
- Possible domains can make use of theories external to the domain in their interpretations.
- Possible domains must proceed by *debate* not only about the facts and the interpretations but also about the theories used in interpretations. Thus, if there is a domain in which there is no possible debate about some claims, an ASP will not apply. This may be the case with mathematics since mathematical conclusions are based on proofs (precise argument-1s) which are often amenable to only one conclusion. On the other hand recent work on the suitability of mathematical proofs as a means of showing program correctness has shown that this view is

over optimistic (De Millo, Lipton and Perlis, 1979).

- Possible domains can deploy their interpretations at *different levels*. The interpreter is designed to take account of the different levels at which a site can be discussed. In the case of archaeology these are, for instance, areas within a site, the overall site, its economy, the social organization of its occupants, the mechanisms for trade, the mechanisms for culture change. The levels of possible reasoning in archaeology are illustrated in Figure 7.1 in which the idea, that interpretation proceeds by argumentation at the different levels using theories or common-sense knowledge, is illustrated.

7.1 Literature

As I have suggested, the approach embodied in ASParch could be applied to other disciplines in the humanities. For instance an ASP could be created for literary criticism (ASPlit). The features which literature and archaeology have in common and which make this possible are given below.

- They are *interpretative* disciplines. The interpretation of a site is not radically different from that of a poem, a novel or a play. Umberto Eco (1985) suggests that the generation of many possible interpretations is one of the jobs of a novel in discussing his own novel, *The Name of the Rose*, when he speaks of, '...a novel, which is a machine for generating interpretations...'. The interpretation of an object is something which can be expressed as an argument and can, in turn, become the object of argumentation. There are other literary critics who suggest that interpretation should not be the main aim of the literary critic. For instance, Culler (1988: 288/9) suggests that literary critics should move towards the study of interpretations as data and away from interpretation as a goal:

Study of the interpretive moves illustrated in both the tradition and our response to that tradition would show how genres, such as lyric, are sets of reading strategies for making sense of language, ways of convincing ourselves not only that language is meaningful, that it will give rise to an intuition or understanding, as would be amply illustrated by our interpretive examples, but also that this is an understanding of the world.

Nonetheless Culler still accepts that the 'present goal' of literary criticism is 'the production of new interpretations of literary texts as the aim and the test of literary

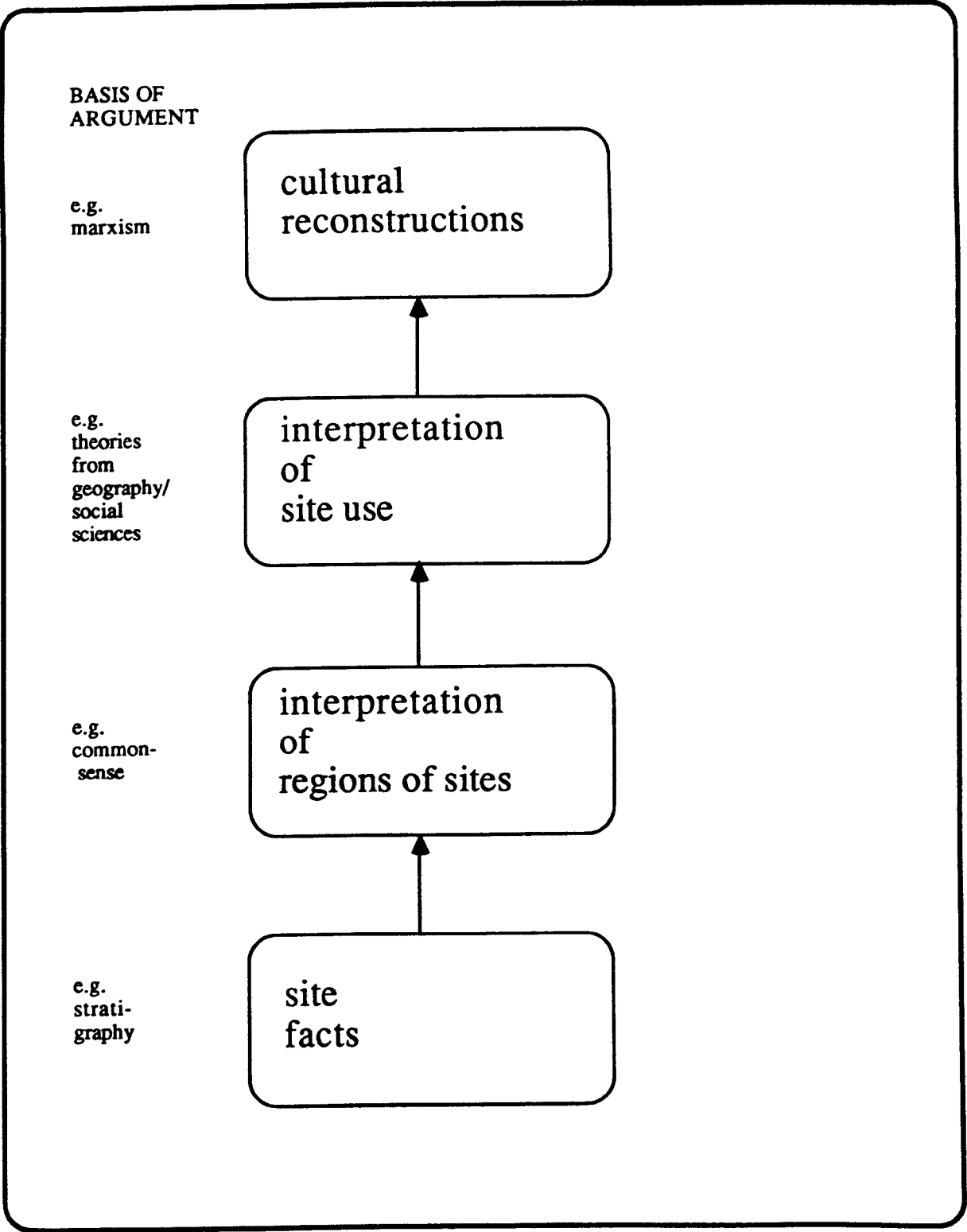


Figure 7.1 - Levels of reasoning in archaeology

studies.' Of course the interpretation of a site is different from that of a book in one sense: The transformations which are applied to the literary data are different from those applied to the site. The aim is similar; to produce a higher level, meaningful and integrated description. This convergence is shown in its extreme form in the title of Hodder's 1986 book - *Reading the Past*.

- They appeal to *theories internal to the domain* in their interpretations. In archaeology a range of theoretical positions is available from the laws of stratigraphy to Binford's middle range theory. In literature there are, for example, the rules of prosody and reader response theory.
- They can appeal to theories *external to the domain* in their interpretations. For example structuralism and marxism provide useful tools in discussing sites and poems.
- They proceed by *debate* not only about the facts and the interpretations but also about the theories used in interpretations.
- They can deploy their interpretations at *different levels*. The notion of levels in archaeological reasoning can be compared to Olsen's notion of the 'hierarchy of levels' in literary criticism (Olsen 1978). In other domains in the humanities, such as comparative religion, the notion of levels is best replaced by the notion of a set of intersecting dimensions of interpretation.

The main use of the computer in literary studies has been for statistical research (e.g. Burrows 1987). More recently it has been suggested by Potter (1988) that the computer can also be used in literary criticism. Indeed she suggests that computing offers 'evidence, precision of measurement and universally-accepted standards of validity' (1988:93). The sort of computing here is akin to statistically oriented approaches. Closer to our own approach, Corns and Smith (1987) posit the computer modelling of theoretical positions (seen as structured bodies of propositions):

Another use of expert systems would be to construct a knowledge base that would enable the user to explore whether their critical assumptions at any juncture were primarily structuralist, reader-response, materialist, etc.

Miall (1986, 1988a) has also proposed the use of expert system technology as a

means of modeling and teaching response to literary artefacts. In a recent conference paper (1988b) he suggests that the proper use of the computer is as a tool which facilitates the student's own research. As a component of this, some capacity for the analysis and storage of arguments would be necessary. At the same conference Slatin (1988) suggested the use of hypertext for the teaching of poetry. This is similar to the work of Catano (1979) on modeling debate about poetry by computer.

It thus seems that *a priori* there are good reasons for suggesting that literature is a possible area of application for ASParch. The existence of work which is similar to ASParch in this domain supports this. In the work of Olsen referred to above there is a well worked out scheme for literary criticism which bears close resemblance to the notion of interpretation we have adopted from Gardin. In Olsen's work (1978) the process of interpreting the literary work is described as follows (ibid:83):

For each segment [or chunk of text] singled out for attention, the reader provides an interpretative description. It is a way of thinking about the segment which brings it into a class or a category. The segment is brought into relationship with the other segments either by the fact that the same interpretative description applies to several segments, or by the fact that the different interpretative descriptions under which different segments have been subsumed can themselves be subsumed under a new interpretative description. If all the segments of the work have been subsumed under some description, and some segments have been interrelated in this manner, it is possible to interrelate more descriptions by introducing a new set of descriptions bringing the first level of descriptions into more general categories. This process of *redescription* may be continued until all the segments of the work are interrelated by being subsumed under some description which also subsumes all the other segments. The single interpretative description becomes a part of an explanatory grid of concepts which covers the whole of the work and relates different parts of the work together... This process can be repeated until all the segments are interrelated in a network of descriptions which articulates the 'meaning' of the work.

This approach can be illustrated by Olsen's treatment of Mack's interpretation of *Hamlet*.¹ A schematic diagram of the resulting interpretation is given as Figure 7.2. In this the lowest level redescription involves the redescription of various chunks or segments of text as, for example, 'The famous questions'. At a higher level these are redescribed as 'symptoms of the mysterious quality of Hamlet's world'. At the highest level the redescription is that 'Hamlet's world is inherently mysterious'. My main point in including this diagram is to draw attention to the similarities between

¹In another context Winston (1980) has made use of Macbeth and Hamlet as test cases for his theory of analogy.

(Redrawn from Olsen, 1978, page 93)

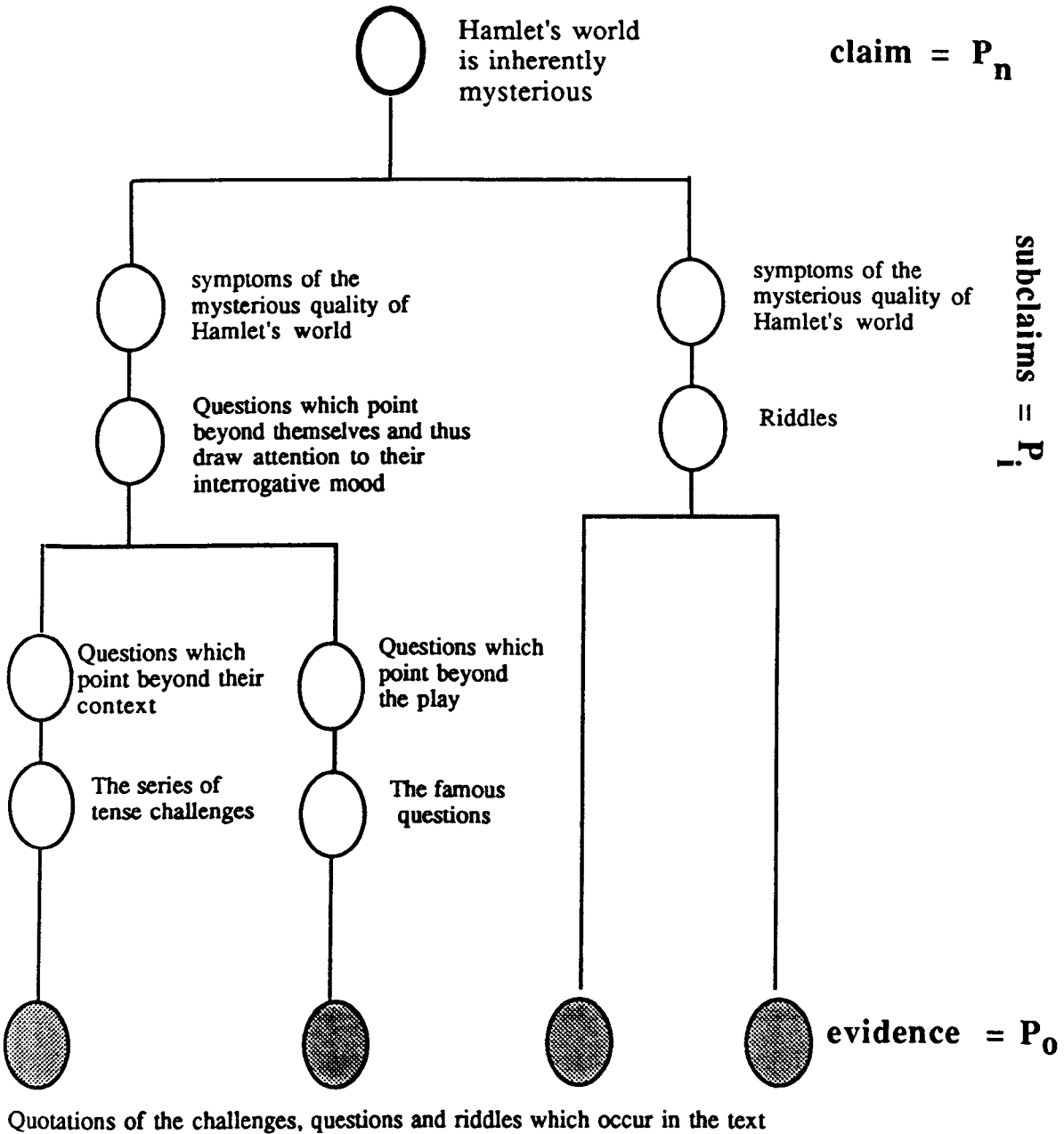


Figure 7.2 - Olsen's diagram of an interpretation of Hamlet

this approach and that of Gardin in archaeology. To this end I have indicated which parts of Olsen's interpretation represent Gardin's Po, Pi and Pn propositions. I will present an ASP-like debate based on this example below.

The approach of Olsen is well-suited to representation in expert system terms. The facts, in this case, are the segments of text and the rules are the rules for redescription of these segments and of other descriptions. At the same time the approach is lacking in four ways:

1. The rigour which is imposed by the computational approach is lacking. For instance, there seem to be large gaps between some of the levels in the Hamlet example.
2. Olsen seems to think that a work can have only one meaning - 'some description which also subsumes all the other segments'. This is unlikely. Indeed our whole approach is predicated on the assumption that, because interpretation is uncertain and multifarious, more than one interpretation of a work will result. Competing interpretations lead to argument and debate. The whole history of Hamlet criticism should make this clear.
3. The sorts of relations possible between nodes at the different levels are not clearly stated. It is not enough to say that 'Redescription is constitutive of a reader's response to a literary work' (ibid:89). There are different kinds of redescription. Olsen accepts this to some extent. On page 95/6 he suggests that redescriptions can be made plausible in three ways: by being sanctioned by the reader's background (non-literary) experience; by following from some notions introduced by the author; and, by depending on 'a shared knowledge of literary practice and conventions'. Nonetheless we need to add to the Olsen account some of the non-deductive transformations which have been discussed above.
4. There is no attempt to deal with the use of theory. The mention of the shared background of common sense and of literary knowledge is as near as Olsen gets. However, I would argue that the account given in earlier chapters of this thesis is a clearer means of relating theory to interpretation.

In Figure 7.3 I illustrate an exchange over the Olsen claim. I have reproduced only part of the Olsen interpretation but enough to show how ASPlit would deal with this

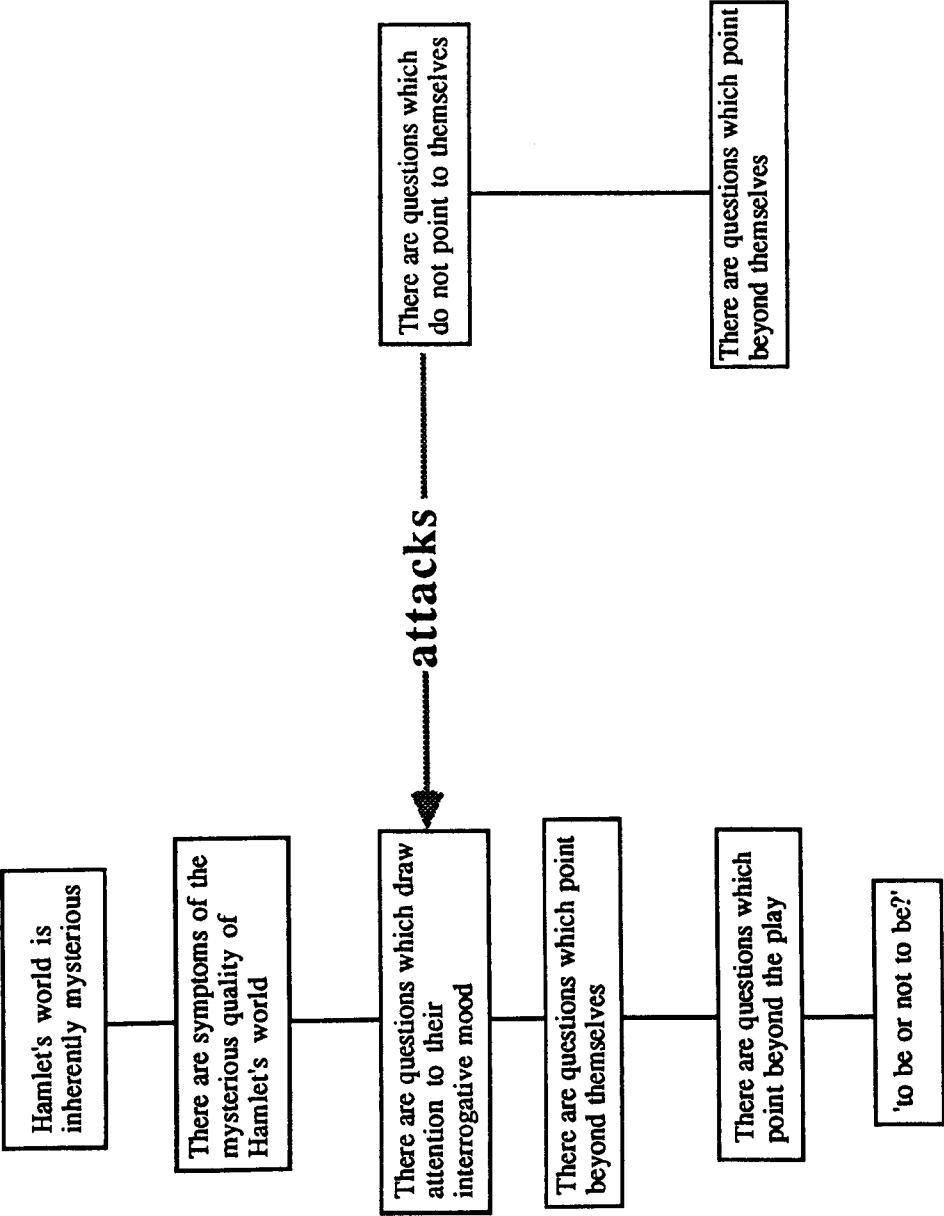


Figure 7.3 - A possible exchange over Olsen's interpretation

case. As the response shows the system might well respond by rejecting the claim that *Questions which point beyond themselves* are therefore *Questions which draw attention to their interrogative mood* on the grounds that an X which points beyond itself does not point to itself.

The method of analysis provided in an ASP seems to provide a better way of dealing with an interpretation such as the above. The analysis is based in a well-founded theoretical approach to argumentation (that of Toulmin) and lends itself well to graphical reproduction on a computer. In addition, an ASP would provide a means of graphically representing exchanges of interpretations (or argument-1s). Finally, ASPlit would provide a means of interacting with the views of previous interpreters and of testing current interpretations against these. This accords well with the notions put forward by Newton in a recent book (1986). According to a review by Ian Saunders (1987) Newton recommends 'an interpretive practice which frankly admits that its own coinage is relative power, not truth, and *engages* with other interpretations. Interpretation ought not to be atomistic but, as Newton puts it, 'agonistic'.

7.2 Legal reasoning

We have discussed the nature of legal reasoning above in our formulation of the tripartite model of argumentation which, following Toulmin, derives largely from a legal model. Given this it seems *prima facie* likely that an ASP could be applied to legal reasoning.

As Dworkin (1982) suggests, legal reasoning can also be seen as a matter of interpretation. This interpretation can be construed as a generator of arguments using diverse forms of inference. It is commonly held that legal reasoning is largely reasoning from precedent i.e. reasoning by analogy (Adam and Taylor, 1987). However as MacCormick (1976) point out, some legal arguments depend on deductive reasoning. Thagard (1988a, forthcoming) suggests that, apart from deduction from legal rules to particular cases and analogical reasoning in the use of precedent, legal reasoning also makes use of explanatory inferences from events to what happened. Thus a lawyer will attempt to show that explanation E explains the evidence ED; a sort of reasoning typical in science and everyday life.

Apart from this, as MacCormick (1978) shows, judges frequently make use of appeals to principle in their judgements. He cites the case of the use of the principle

(ibid: 163, quoting Lord Macmillan) that 'The duty to take care is the duty to avoid doing or omitting to do anything the doing or omitting to do which may have as its reasonable and probable consequence injury to others'. This principle can be restated in less legalistic language as follows: We have a duty to avoid actions or omissions which we could reasonably expect to result in injuries to others. MacCormick also points out (ibid: 161) that 'no clear line can be drawn between arguments from principle and from analogy'. It is often the case that a principle can be used to determine whether an analogy is legally relevant (a point similar to those made by Hodder with regard to relational analogies and discussed in chapter 4). 'So the relevance of the analogy is dependent upon perceiving a rational principle within which the two items compared can both be contained...' The above principle, for example, allows the comparison of attempts to rescue property with attempts to rescue life insofar as there is a duty to prevent circumstances which could lead to an attempt to save property or rescue life with the possibility of injury to the salvor or rescuer.

Dworkin also suggests that the law progresses by what Newton calls 'agonistic' interpretation. He argues against the notion that interpretation in the law (and indeed in literary criticism) is purely objective (legal judgments follow inexorably) or purely subjective (just made up). He suggests that legal interpretation is like the case of a group of novelists who are set to write a novel. Each novelist has to provide an extension of the novel which flows naturally (and indeed, logically) from that of earlier novelists. In the same way legal reasoning is a "chain enterprise". Results are not simply given. However they depend on the previous decisions and practices of lawyers and judges (or of critics) rather than personal whim. ASPlaw might function as a means of making explicit some of the results of these practices and how they bear on current decisions. This assumes that the law (or other forms of interpretation) is not simply a matter of fashion. On the other hand, insofar as some of these practices, conventions and so on (or the understanding of them - the internalized awareness of the tradition) remain tacit and cannot be articulated, an ASP can only act as a partial aid.

While legal reasoning seems to be a good domain for the use of the techniques discussed in this thesis, there has been much debate in recent years about the feasibility of using computational models of legal knowledge to decide legal questions. It may simply be impossible to use computers to produce such decisions. Dworkin himself holds this view (1982) - 'There is, of course, no algorithm for deciding whether

a particular interpretation sufficiently fits that history [legal history] not to be ruled out'. Sergot, Sadri, Kowalski, Kriwaczek, Hammond and Cory (1986) suggest that it is possible to formalize in a computer program statute law (the specific example is the British nationality Act of 1981) and to use this 'to determine whether in a given circumstance a particular individual is or is not a British citizen' (ibid: 377). Leith (1986: 545) has argued that the work of Sergot et al. is 'a dangerous oversimplification of a research field'. He suggests that there is no such thing as a clear rule of law and that therefore the use of expert systems to model these is invalid. Leith argues that 'the legal process is principally a process of negotiation' (ibid: 549) and draws attention to the use of 'ouster' clauses in legislation. These are inserted by the legislature to control the judiciary but are frequently overruled by judges. The effect is that, looked at from the point of view of deductive logic, there are contradictions embodied in legislation. For example one clause may suggest that the decision of a government official cannot be appealed to by a court while another provides a general right of appeal.

Lying behind this debate is a confusion about the exact nature of legal expertise. If Sergot et al. are only attempting to provide a tool which derives the logical implications of certain clauses in an act then Leith should have no quarrel with them. Insofar as they attempt to portray this tool as capable of making decisions then they are wrong and Leith is right. The logically derived claim can only form part of an interpretation by a lawyer or a judge; it cannot take its place. Judicial or legal knowledge is not simply the logical deductions from a set of clauses in a statute book. It includes at least the knowledge of other statutes and cases, and the ability to reason from these. More than this, it includes the ability to assess evidence, produce causal models of situations (if Thagard is right), take account of the mitigating circumstances and interpret the legal statutes in a manner free from political interference but nevertheless in accord with generally accepted social norms.

An ASP cannot take the place of a lawyer or judge but it may provide a tool which can store in a structured form the exchanges between lawyers and provide for trainee lawyers a means of interacting with great arguers of the past.

Thus in Figure 7.4 we can see an argument adapted from the example from MacCormick discussed above with an appropriate response. In this case the argument is that the widow of X should get compensation because:

- (a) he was killed in attempting to avert an accident involving a train which was

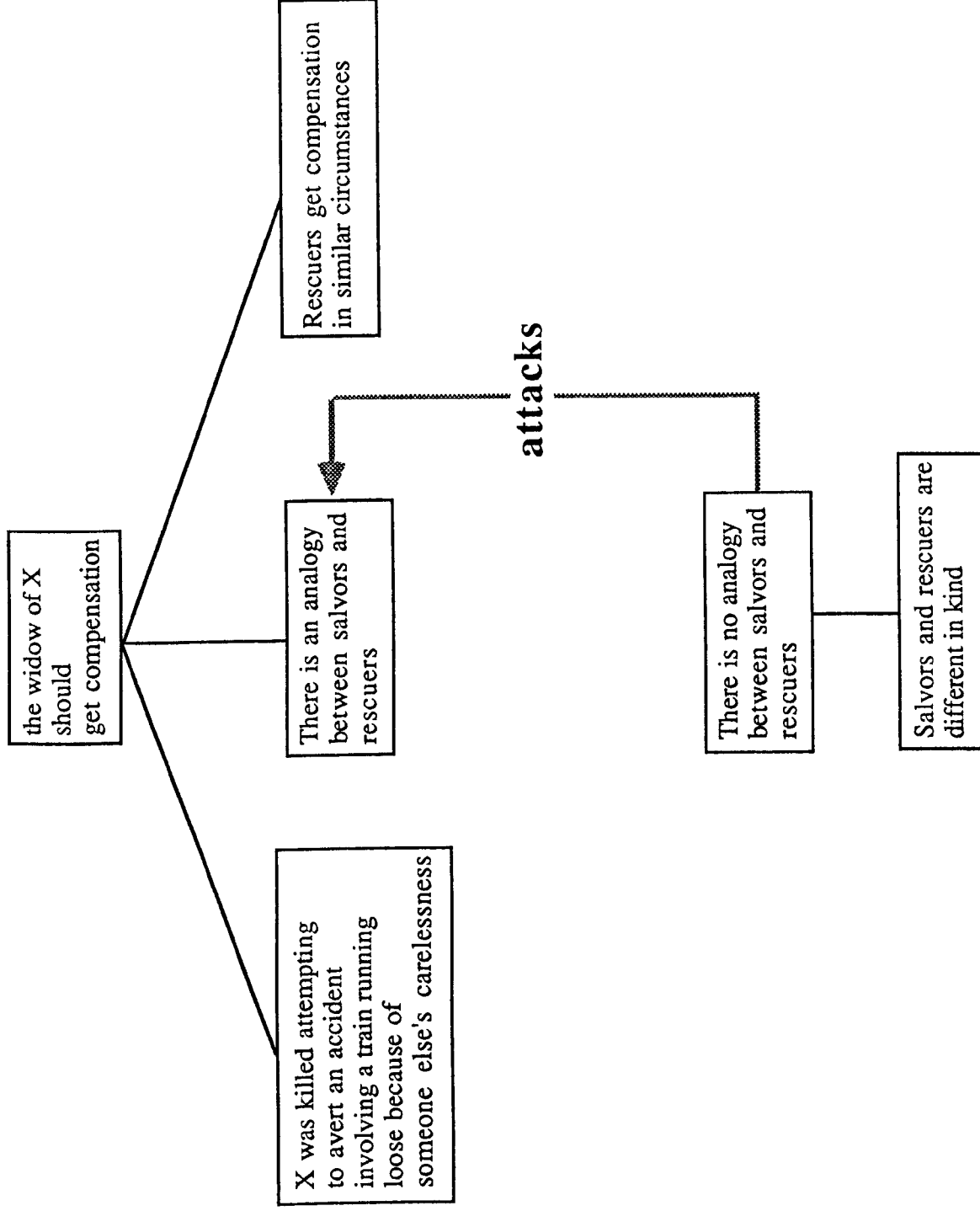


Figure 7.4 - A legal argument

running loose because of someone else's carelessness;

(b) 'It was settled in Scots law that a rescuer's act in saving another person endangered by the wrongful act of a third person is entitled to reparation from the latter if he suffers injury'

(c) There is an analogy between rescuers of life and salvors of property.

The ASP-like analysis shows that C1 is supported by (a) and (c). Item (b) above is treated as a ground here though it might more properly be dealt with as a warrant. In this latter case the reasoning would be more complicated with the analogy acting as ground for the use of (b) as a warrant in this particular case. I have illustrated the former for simplicity but it should be noted that there is no theoretical reason why a warrant should not be argued about using an ASP. The counter-argument is based on the position taken by Lord Mackay. He rejects the analogy on the grounds that a rescuer is different from a salvor which in turn depends on the ground that there is a difference between human life and property.

Again as with ASPlit, ASPlaw provides a means of analysing and displaying both individual argument-1s and the argument-2 which they constitute.

7.3 Darwinian theory

In a famous debate in 1860 at the British Association 'Soapy Sam' Wilberforce, the Bishop of Oxford, confronted T.H. Huxley in a debate about Darwinian theory. The Bishop lost decisively, thus allowing the shift to the Darwinian paradigm which underpins modern biology.² This shows the importance of debate in the evolution of scientific worldviews.

It would be interesting to attempt to implement the debate between Huxley and Wilberforce. However by way of an example I present the following argument from Darwin's *Origin of Species* with a putative reply. I have, so far as I can, attempted to analyze the argument in a Toulmin-like manner. The following deals with only a small part of the overall argument of the book. Interested readers can turn to Thagard (1988a, forthcoming) for an examination of the overall theory in the context of an

² In a famous incident in a later debate (see Huxley and Kettlewell, 1965) Wilberforce asked Huxley whether he claimed descent from a monkey through his grandfather or his grandmother. According to Huxley and Kettlewell, Huxley replied to the effect 'that if he had to choose between a poor ape for an ancestor and a man highly endowed by nature and of great influence, who used his gifts to introduce ridicule into a scientific discussion and to discredit humble seekers after truth, he would affirm his preference for the ape'.

analysis of its explanatory coherence.

Darwin argues for the following conclusion in chapter 1 of the *Origin of Species* (Darwin, 1979):

(C) "From these several reasons, we may conclude that all our domestic breeds are descended from the rock-pigeon with its geographical sub-species."

His argument (in a much simplified form) has the following grounds (where Gn = a top-level ground, SGn = a ground in a sub-argument and W = a warrant):

(G1) One original is adequate.

(SG1) *Columba Livia* has required characteristics

(G2) Only Rock Pigeon possible

(SG2) only rock pigeon has correct colouring

(SG3) when breeds crossed rock pigeon colours appear

(SG4) there is reversion to ancestral characteristics of rock pigeon

(W1) well known principle that reversion shows ancestry

(G3) Only one ancestor

(SG5) Mongrels all perfectly fertile

(W2) most hybrids of two species are infertile

The counterargument suggests that ground G1 is erroneous:

(C2) One original is inadequate.

(SG6) 7 original stocks needed.

(SG7) 7 needed for all different breeds.

In fact this argument is considered by Darwin himself and rejected using the following argument:

(C3) Not 7 originals

(SG8) hard to breed even one species under domestication

(SG9) unlikely primitive man domesticated 7 species.

The graphical form of this is shown in Figure 7.5. We can see from this example that the ASP methodology is particularly useful for the automatic display and interaction with argument-2s. As I have said Darwin's argument proceeds by first putting forward counterarguments and refuting these. Apart from the methodology, an ASP represents a computational aid in this process of arguing with oneself and/or producing lengthy argument-2 structures.

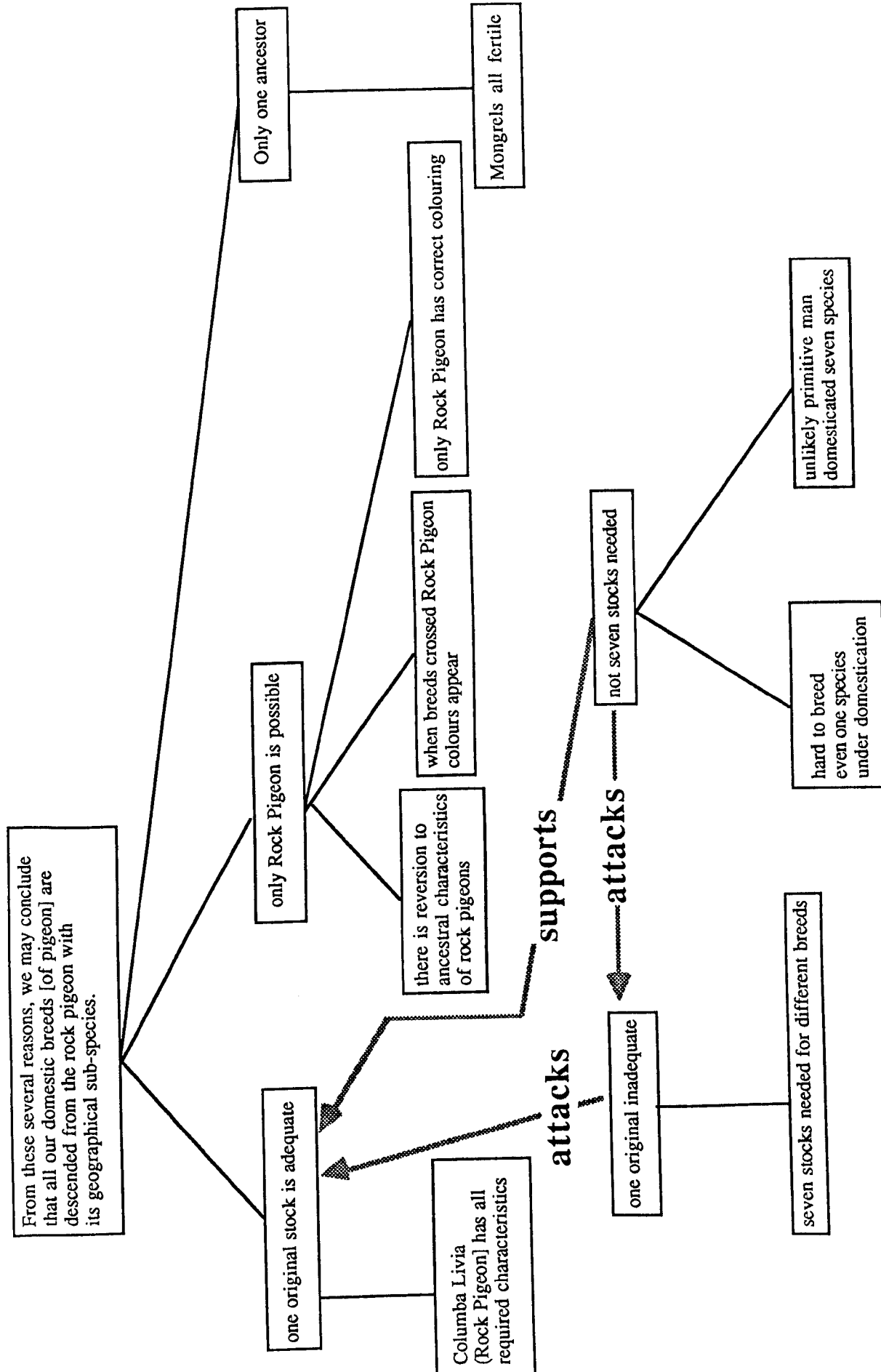


Figure 7.5 - An argument from Darwin

7.4 Conclusion

In general terms the progress of any science can be seen in terms of a debate which never reaches a conclusion (i.e. there are never any final answers) but which, for a time, is satisfactory. In Figure 7.6 I present an example from the history of science surrounding the dating of the earth. Where Bishop Usher of Armagh had worked out from the Bible that the date was 4004 BC, Lord Kelvin showed in 1862 that the date had to be much greater, from extrapolations of the rate of the cooling of the earth. He proposed an age of 24 million years. Subsequent research has put that figure back even further (to something like 4.5 billion years). The flaw in Kelvin's argument being that some of the rocks are radio-active and therefore produce their own heat. Possibly new research could serve to put the results back even further. We can think (as Gardin does) of this process as a continual spiral towards a mythical omega point. Notice however that it is a spiral and that the context within which the argument proceeds depends on the prior contexts. Typically the context will get smaller (as scientific disciplines get more specialised). Thus, what counts as a good argument will change as the context of the inquiry changes. Put baldly, the bible is no longer seen as a source of empirical evidence about the origin of the earth. Since Kelvin's time, other discoveries have been made (not least by Kelvin himself) and these now need to be taken into account.

There is another sense in which argument-2s enter into scientific discourse. The above might be termed the macro-argument-2. The micro-argument-2 represents the case where debate among a group of scientists is used as a means of advancing a new theory. This can be illustrated vividly by the case of Crick and Watson's discovery of the helical nature of DNA. Both the book by Watson (1968) and the recent TV dramatisation show how the young scientists continually argued over the details of their models, collected new information, rebuilt models (both physical and notional) solicited comments and acted on the comments. The resulting model stands unchallenged (until some new discovery or reformulation of theory either advances our knowledge to a new stage or completely overturns our understanding of genetics). There is an interesting quote, from our point of view, which illustrates how the Crick and Watson laboratory went about their work. This comes originally from a paper by Platt (1964) which was drawn to my attention by Stefik et al. (1987).

On any given morning at the Laboratory of Molecular Biology in Cambridge, England, the blackboards of Francis Crick or Sidney Brenner will commonly be found covered with logical trees. On the top line will be the hot new result just up

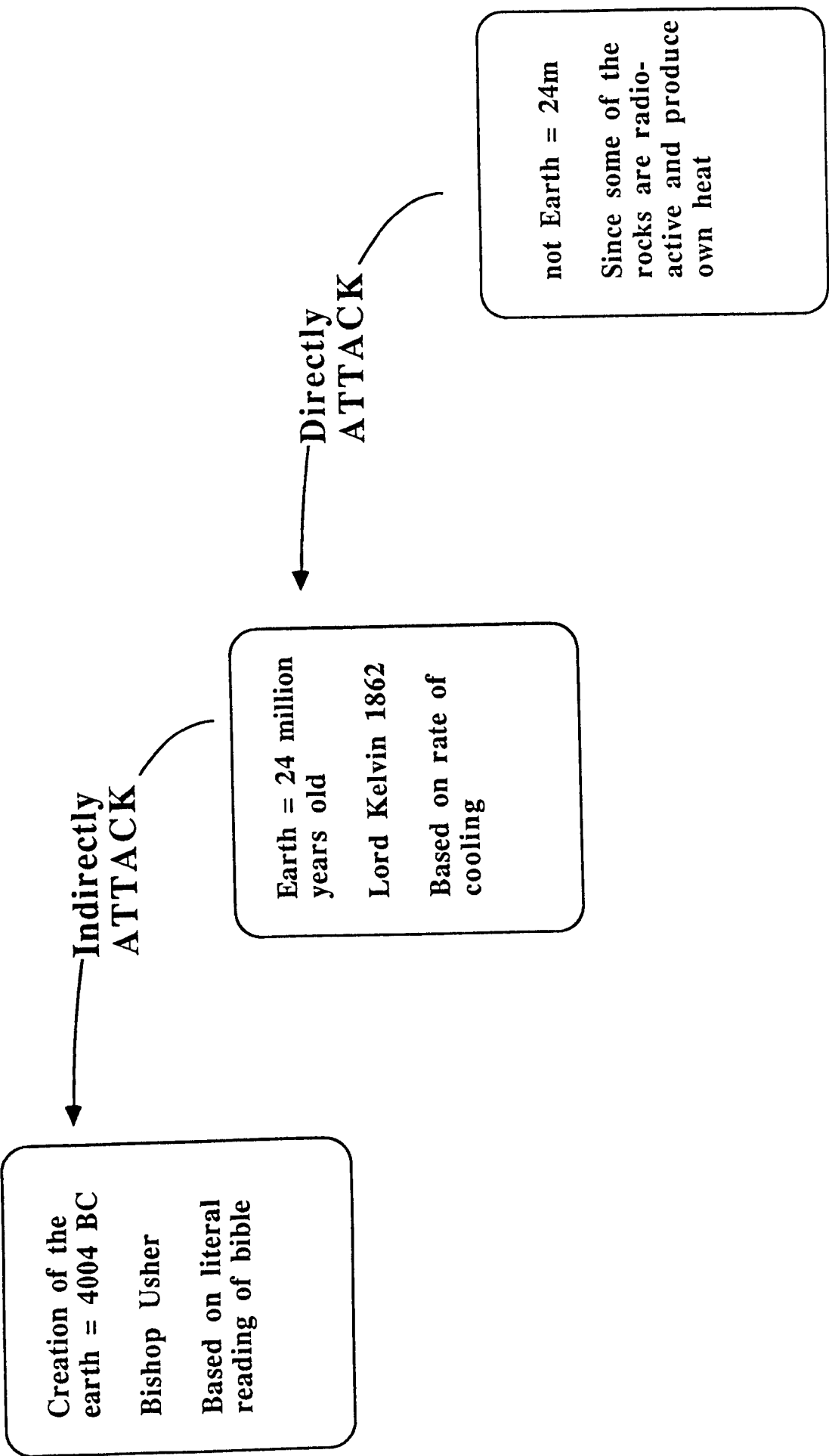


Figure 7.6 - Dating the earth

from the laboratory or just in by letter or rumor. On the next line will be two or three alternative explanations, or a little list of "What he did wrong." Underneath will be a series of suggested experiments or controls that can reduce the number of possibilities. And so on. The tree grows during the day as one man or another comes in and argues about why one of the experiments wouldn't work, or how it should be changed.

I envisage ASParch both as a Crick and Brenner blackboard and as one of the colleagues who suggests and criticises. Thus the computer is doing two tasks for which it is well suited (Design Principle P1 above): (a) storage, retrieval and display in graphical form; (b) formalised inference (albeit using heuristic rules augmented to simulate various forms of plausible reasoning). I would suggest that there are two main processes which occur in scientific disciplines: experiment and debate. The debate is useless without the data provided by experiment. The experiment depends on argument for its proper interpretation and for the necessary design. In other disciplines (such as archaeology) experiment has a minimal role though there is a branch of archaeology which attempts reconstructions of ancient artefacts, buildings and agricultural practices (see Coles 1973). Here the evidence on which argument is based is provided by excavation, historical sources, ethnographic surveys and so on. Argument also aids in the design of excavations and surveys. Archaeology as a science is closer to cosmology where the variables involved cannot be easily manipulated (e.g. temperatures within milliseconds of the creation of the universe). In the law, argument is based on evidence gathered by police and by the questioning of witnesses. In literary criticism the evidence is the text. However, argument, as well as being based on the evidence, can also be involved in selecting the appropriate evidence. As Potter (1988) suggests, the use of computational methods for selecting and manipulating this data can add a greater objectivity to the discipline. Debate will still be required as to which computational techniques to use as well as about the proper interpretation of the results.

As the examples show the ASP methodology is principally important in enforcing a coherent method of analysis, in providing graphical displays of the relationships in argument-1s and argument-2s and in providing an automatic colleague to try arguments out on in an attempt to assess and strengthen them. The above discussion and the hand-crafted examples do not *show* that ASPs can be applied widely. They

only make it more likely. Such proof will only come when a full implementation of the program design is carried out and applied in these areas. The above discussion is only meant to be suggestive. Nonetheless it seems probable that if ASPs work in a single domain they will work in any domain which is similar. A domain will be similar if it the criteria put forward at the beginning of the chapter are met. Of course more needs to be done than to implement the ASP shell. Each domain requires a knowledge base encoded in the standard ASP format and which includes all the intrinsic theories of the domain and any extrinsic theory not known to the system. The initial knowledge base can be created using standard knowledge engineering techniques and augmented during use. However, unless we make use of a system (such as OpEd - see chapter 3) which includes the capacity to comprehend arguments expressed in natural language we must rely on a method of hand-crafting to extract, from texts, the arguments we wish to simulate. This method has been followed in all the examples used in this thesis including those given in the present chapter. It involves:

- the use of the model of argument as a means of analysing the texts and defining the components which are to be expected;
- the use of textual clues (such as 'because' or 'since') as a means of deciding which propositions have which function - Cohen (see chapter 3) includes a theory of clue words in her computational model;
- the use of the semantics of the contents of the propositions as a means of determining the relations between parts of the argument-2.

We also need to identify the *argument style* for the domain and hence what sorts of response and assessment rules are needed. Given these the ASP could function as a sort of high level shell into which knowledge bases could be slotted and by means of which stylized argument exchanges about a domain could proceed.

-o0o-

Chapter 8 - Discussion

Argument: While we have, to some extent, succeeded in our three aims there are inadequacies in our approach. There are several possible reasons for our inability to construct a complete arguer. Nonetheless, in one possible development, ASPs can provide a basis for a mixed AI / Hypertext environment for promulgating arguments.

This final chapter is intended to summarize and assess what has been achieved and to suggest improvements and directions for research which might build upon what has been achieved. I will start with a section summarizing the results, then discuss some of the failings of the approach and close with some thoughts about future directions.

8.1 What has been achieved?

We have presented a design and implementation of a computer program which is intended as a means of supporting arguers from archaeology, and, as we have seen, other areas of the humanities. In brief, this program can:

- provide argument-1s about limited domains
- respond to user argument-1s about these domains
- display user and system argument-1s in a graphical form

By these means the system:

- provides a means of communicating knowledge about a domain
- provides a means of producing and testing argument-1s
- reveals the grounds for pieces of archaeological reasoning
- acts as an informal tutor for a method of analysis
- acts as an informal tutor for domain-specific responses

The program provides assistance in at least three ways: (a) ASParch enforces a method of analysis on the user; (b) ASParch displays in a diagrammatic form the arguments produced by users and the system; (c) The program provides a rudimentary means of interacting with the machine in relation to the argument.

(a) An ASP provides more than the objectivity and clarity given by the use of computers *per se* - it provides a method of analysis and a language for arguing. Indeed the benefits accruing from the objectivity of the machine may be exaggerated. It has been argued that in fact the computing machine embodies a particular ('rationalist') view of thinking. For instance Hunter (1988:19) has recently argued against a misconception she identifies - '...that the computer is only a tool and that therefore its use is not subject to interpretation...' She also points out (ibid) that '...most modern science and technology constructs its representations of the world through a rational analytic logic...In contrast, humanities teachers often use dialectic, or overt metaphorical and analogical systems to produce representations.' Thus it is necessary to extend the machine so that it can cope with 'real' arguments from users in the humanities. Not only because it needs a formal language for arguments before it can deal with them but also because we need to circumvent the hidden metaphors which the machine contains while seeking to impose as few of our own as we can. However, because all formal languages will contain hidden assumptions, the 'arguing' machine must go further than the provision and evaluation of an argument language. It must interact with the user in order for *its* own assumptions to be tested.

(b) The diagrammatic method of displaying propositions is common in the kind of argument analysis which informal logicians employ. Toulmin (1979), Scriven (1976) and Fisher (1988) all employ some diagrams. Thus the method is natural. The use of advanced computer graphics allows the possibility of interaction with the diagram (requests for further information, responses to particular propositions) as well as the 'animated' display of an entire argument. The use of computer graphics also renders vivid the contextual nature of inference and argument. Thus while an argument may take place in an overall context of beliefs and theories, individual elements in the argument may depend on entirely different contexts. An argument in a law-court may depend on a piece of physics. An argument in an artificial intelligence thesis could include an appeal to common sense. Whatever position we adopt on the question of how we assess such arguments (using the techniques of the overall field or those of the specific field) there is at least something to be gained by showing how an argument is composed of these differing elements.

(c) Interaction is possible because of advances in artificial intelligence which allow the encoding of domain knowledge as well as the representing of insights drawn from psychological studies of how arguers respond to arguments. As we have seen, there are three aspects to this process: firstly, the system must be able to create arguments (i.e. follow a methodology for argument making which fits with the method of analysis); secondly, it must interpret or analyse user arguments and assess their weaknesses; and, thirdly it must come up with a satisfactory response. Note that the method of analysis and assessment is not simply the methodology of formal logic by another name. We have attempted to render in a computational form some of the insights and methods of students of *informal* logic. Thus we have not presented a method of analysis and assessment which depends on the logical *form* of a piece of argumentation. The meaning of the propositions is taken account of at least to the extent that the meaning is determined by a particular context (of facts and operations) which differs from domain to domain. This leads to a second major divergence from formal logic. In everyday arguments, and in some fields more than others, non-deductive methods of inference are employed. In the above we have concentrated on the use of analogies.

As well as the design of a tool to support arguing we have also shown how this new approach can deal with inadequacies in the explanation capabilities of expert systems insofar as explanation (or, at least, justification) finds a natural place within the context of argument. At the same time we have produced a model of argument which clearly distinguishes the different components of argument, the levels of argument and how they are interrelated and how argument is related to interpretation and explanation. This model has been substantiated to some extent by the implementation of a program design derived from it.

Furthermore, we have gone some way towards replying to the worries of domain specialists such as archaeologists as mentioned in chapter 1 above. If we reduce these worries to three main items we can more clearly see to what extent they may have been quelled. The three main worries are:

- (1) The *formalization* problem i.e. an inadequate model of an area of archaeological expertise may be used.

(2) The *fossilization* problem i.e. the domain knowledge contained in the expert system may come to be regarded as fixed and complete.

(3) The problem of the *partially non-deductive* nature of archaeological reasoning.

Item (3) is dealt with insofar as the program is able to deal with analogical reasoning in archaeology since this seems to be the main non-deductive form of reasoning in this field. Items (1) and (2) are not dealt with by the implemented program. However the model allows for the system to assess an argument and as a result change its knowledge base. Thus the knowledge is not fossilized and can be changed by the user either by changing the knowledge base directly (entering new rules) or during an argument exchange. Since the system is capable of dealing with the theoretical backing for the warrants or rules used in reasoning the danger of the knowledge base builder imposing an alien model on the field is minimized. The formalization problem is trickier since any formal system can be objected to in this way. Given that the system can be changed during use and given that there is a reasonably flexible data structure for storing knowledge, this difficulty is again minimized.

Apart from the main goals of this thesis there are two subsidiary benefits of the ASP approach in that this approach provides a means of dealing with two important problems in expert systems research: knowledge acquisition and uncertainty.

Knowledge acquisition

As well as problems to do with explanation, another major problem in the development of expert systems has been the acquisition of knowledge from domain experts. For archaeologists this manifests itself as the formalization and fossilization problems mentioned above. The knowledge acquisition problem arises because of the need to transform the informal expertise of the expert into the sort of formalism required for a computer program to be able to function. There have been many approaches to this problem (Wielinga and Breuker 1986, Bylander and Chandrasekaran 1987, Gammack and Young 1985, Boose 1985, Boose and Bradshaw 1987, Eshelman, Ehret, McDermott and Tan 1987, Clancey 1985a). The ASP approach can be viewed as an integrated approach to explanation and knowledge acquisition. It has similarities to the system described by Kim and Pearl (1987) which integrates knowledge acquisition and advice-giving in the situation assessment domain. It is also similar to one of the more interesting of these systems, in the

context of the present thesis, Mole, which is produced by McDermott and his colleagues (Eshelman et al. 1987). Mole is a performance program as well as a knowledge acquisition system. Basically the system works by comparing its analysis of a problem (to do with a steel rolling mill) with that of the expert. If there is a disagreement then the system sees if there is a path from what it knows to the expert's solution. If there is, Mole sees if it can re-evaluate the support values so that this becomes the favoured solution. Failing this, it asks for more supporting information. If there is no path to the solution then there are two possibilities: either a hypothesis was rejected when it should have been accepted or accepted when it should have been rejected. If the former, then the system asks the user for some extra symptom which needed the support of the hypothesis. If the latter the user is asked for some other hypothesis to explain the symptom explained by the accepted hypothesis. This system incorporates some ideas which are very similar to those in ASPs in that it acquires its knowledge from the user in an argument-like manner.

An ASP can be said to acquire knowledge in a Mole-like manner when, in the course of an argument session, the user puts forward evidence and/or warrants which are new to the system. If the system finds no means of combatting this evidence (or its means are weaker than those of the user by some assessment procedure) then the system will have acquired new knowledge. Not only that but it will have acquired it in a natural exchange and in a manner which allows it to be readily assimilated into the rest of the knowledge base in that the system will have acquired with the knowledge various links between it and other knowledge. The system, insofar as it can lose an argument or debate, may also change its mind about its own knowledge. Thus new knowledge will be acquired and old knowledge will be continuously updated and revised as a result of argument exchanges. This process is akin to that which we all experience in our acquisition of knowledge about the world. This should not be surprising since, as I have suggested above, one of the principal points in argument exchanges is an epistemic one. They are a means of communicating and acquiring knowledge. Apart from the naturalness of this means of interaction an ASP scores over a system like Mole in that the ASP has a fully worked out model of argumentation.

Uncertainty

As we have seen in chapter 4, there are two forms of uncertainty in archaeological reasoning:

- a) That which arises because more than one interpretation of the data will always be possible;
- b) That which arises because the data is incomplete.

The standard means of resolving these sorts of problems include various conflict resolution methods, certainty factors and fuzzy logics (e.g. Shortliffe, 1976, Zadeh, 1988). These methods do not lend themselves readily to the provision of explanations, justifications or arguments for the system's conclusions.

One further technique for dealing with uncertainty which is more closely related to our own is that of Paul Cohen (Cohen, 1985, Cohen and Grinberg, 1983) in which he takes an endorsement based approach to uncertainty. In Cohen's approach, endorsements or explicit records of reasons for belief are attached to rules and facts and propagated by the system during reasoning. The endorsements are used to decide between conflicting possibilities. In Cohen's words an endorsement is a record 'for believing or disbelieving a hypothesis' (Cohen et al, 1983:355). Using this approach different types of evidence can be treated in different ways and weighed differently in different contexts. This is comparable to the different sorts of argument type which make up an argument style. As Cohen says, 'it makes little sense to speak of certainty except with respect to a task' (ibid: 356). Cohen shows how this technique improves upon the standard probability based methods in his analysis of an argument to do with hominid origins in Africa.

In chapter 4 we saw how the expectation-based approach to modelling interpretation could be used to reduce uncertainty. Here we shall briefly indicate how the full ASP model can be used for this purpose.

Cohen's theory differs from my own in that there is no propagation of the reasons for certainty values in my approach. The reasons for X being held are stored locally in an argument node. Assessment procedures can access these as necessary. Thus it is unnecessary to propagate *all* the endorsements or reasons for believing X. I would go even further and suggest that the best way to deal with uncertainty in archaeological reasoning (and other forms as well) is to regard the conclusion as

provisional; as satisfactory for the moment. At any future point new evidence may cast doubt on the conclusion or conversely add strength to it. This sort of rational revisability requires a system which can deal with argumentation - an ASP - so that reasons can be associated with conclusions, arguments can be weighed against each other and knowledge can be updated and revised.

8.2 Weaknesses in the approach and some possible reasons for these

I have already given details of the inadequacies of the implementation of ASParch in the final section of chapter 6. Here I am more concerned to highlight what I take to be inadequacies in the overall approach.

- While ASParch produces argument-1s, the question arises of the relationship of these to real arguments. They differ in many ways. For instance real arguments are wide-ranging, subtle and complex. This inadequacy is partly explained by the incompleteness of the knowledge bases used in the above examples. This can be remedied. Nonetheless the arguments produced will still seem unrecognizable to some users.
- The sorts of response produced don't seem always to the point. Nor do they reflect the idiosyncrasies of an individual's response.
- The influence of theoretical, ideological and other viewpoints is more complex than that incorporated into the program.
- There is much more to analogical reasoning than the simplified approach adopted suggests.
- ASPs do not deal with natural language and have no knowledge of the structure of dialogues. Both are needed before the program becomes a usable tool. Without them the arguments seem stilted and unnatural.
- While the approach allows for multi-dimensionality of arguments there is still a great deal of inflexibility in how arguments are input, output and represented. This may be inevitable given the need for computational tractability but it narrows the range of possible arguments.
- If we take the design principle P1 seriously then it seems wrong to have any form of assessment in the machine. This should be left to the human user while the system presents all the possible arguments.
- Apart from the fairly *ad hoc* methodology for extracting arguments from textual sources mentioned at the end of chapter 7, there is no real theory of how this

should be done. As we suggested in chapter 1, written arguments can be interpreted in many ways and each interpretation can be argued for or against. While the ASP methodology provides a means for acquiring the complete structure of an argument during interactive exchanges it does not provide a means of acquiring knowledge from texts. This is the function of natural language programs such as OpEd. It may, however, be possible to extend ASParch so that it can choose between multiple interpretations of an argument as given in a text (using a more computationally precise set of procedures based on those given in chapter 7) and hence acquire knowledge from texts as well as from users.

I will now attempt to suggest reasons for some of the failings mentioned above. I have identified nine main difficulties which lie behind the above failings. Some of these were apparent at the outset, others have only become clear through making the attempt to model an arguer. Furthermore, some are generally applicable to AI as a whole while some are only applicable to the modelling of argumentation. The difficulties are:

- Inability to capture all the background necessary for arguing.
- Argument is inherently metaphorical.
- Argument is culturally relative.
- Argument includes tacit elements.
- Argument touches on value systems.
- Arguing is not something done in a cold, unemotional manner.
- Arguments are never divorced from personal, social and political considerations.
- Arguments are creative.
- Interpretation is not simply the application of transformations to data.

As we shall see as we deal with each of these in turn, there are many interrelationships between the nine issues.

8.2.1 Inability to capture all the background necessary for arguing

This is basically the point made by Winograd and Flores (1986: 74 ff.) which we have alluded to already. As they say:

...our openness to experience is grounded in a pre-understanding without which understanding itself would not be possible. An individual's pre-understanding is the result of experience within a tradition. Everything we say is said against the

background of that experience and tradition, and makes sense only with respect to it. Language (as well as other meaningful actions) need express only what is not obvious, and can occur only between individuals who share to a large degree the same background. Knowledge is *always* the result of interpretation, which depends on the entire previous experience of the interpreter and on the situatedness in a tradition. It is neither 'subjective' (particular to the individual) nor 'objective' (independent of the individual)... Artificial intelligence is an attempt to build a full account of human cognition into a formal system (a computer program). The computer operates with a background only to the extent that the background is articulated and embodied in its programs. But the articulation of the unspoken is a never-ending process. In order to describe our pre-understanding, we must do it in a language and a background that itself reflects a pre-understanding. The effort of articulation is important and useful, but it can never be complete... At the other extreme [from building systems which deal with a limited articulatable domain] lies the attempt to build systems that allow us to interact as though we were conversing with another person who shares our background. The result can easily be confusion and frustration, when breakdowns reveal the complex ways in which the computer fails to meet our unspoken assumptions about how we will be understood... We must be especially careful in dealing with so-called 'expert systems'. The ideal of an objectively knowledgeable expert must be replaced with a recognition of the importance of background.

Even if were possible to capture all this knowledge it would be an immense task. Although there are recently reported attempts to capture, for example, the whole contents of an encyclopaedia (e.g. the CYC project, Lenat, Prakash and Shepherd, 1986) there is also the question of how this knowledge is processed in the creation and comprehension of arguments. And as the above quote suggests it is the unsaid and the tacit (see below) which ground our knowledge claims and interchanges. Thus I would suggest that even if Winograd and Flores are wrong, aspects of human arguments will always elude an arguing machine.

8.2.2 Argument is inherently metaphorical

In their seminal work on metaphoricity in everyday discourse, Lakoff and Johnson (1980a, 1980b) discuss the metaphorical nature of the concept of argument. They put forward instances in which it seems obvious that we regard an argument as a war. For example 'He *shot down* all my arguments'. Such instances could be multiplied indefinitely. However as they point out '... we don't just *talk* about arguments in terms of war. We can actually win or lose arguments. We see the person we are arguing with as an opponent. We attack his positions and defend our own' (Lakoff and

Johnson 1980:4) They go on, (ibid: 5):

ARGUMENT is partially structured, understood, performed, and talked about in terms of WAR. The concept is metaphorically structured, the activity is metaphorically structured, and, consequently, the language is metaphorically structured. Moreover this is the ordinary way of having an argument and talking about one.

However, as they go on to point out, the metaphor of war, while it may be the dominant one in our culture, is not the only means of structuring argument we have. An argument can also be considered as a journey, a container and a building (ibid 89). Nonetheless, the primary metaphor for argument is war. But war is something humans engage in as part of wider political and national struggles. Thus an argument can be seen as a means of gaining, asserting and defending power. 'Part of being a rational animal, however, involves getting what you want without subjecting yourself to the dangers of actual physical conflict. As a result, we humans have evolved the social institution of verbal argument' (ibid 62). Even academic argument (Lakoff and Johnson call it *rational* argument) is conducted in a warlike manner. 'The tactics of intimidation, threat, appeal to authority, etc., though couched perhaps, in more refined phrases, are just as present in rational argument as they are in everyday arguing and war' (ibid 65).

As archaeologists are well aware, the use of elaborate displays of one kind or another (e.g. the building of Stonehenge) can be seen as one means of asserting and maintaining power. The ostentatious display of a lawyer or a young academic at a conference is another means of gaining or displaying power. Seen in this light, the young academic has much in common with warriors as diverse as the ancient Gaul, the medieval knight and the American Indian. Display and prowess are the dominant themes.

Of course, as is the way with real wars, the notion of an argument can be construed as a game. We talk of war games. People re-enact the famous battles of the Peloponnesian War. However as with war-games, the games we play with arguments are often close to the real things. Soldiers work out tactics and develop skills by simulating the real thing. The school child debating in the class room for the first time is acquiring the skills necessary for a wide range of occupations from lawyer to local councillor. In our everyday arguing a game argument often turns into the real thing when some deeply held belief is challenged¹. Even philosophy, that paradigm

¹ A recent newspaper article with the headline 'Punch-up poet counted out' relates the case of an

of game argument, becomes a battleground strewn with the corpses of defeated arguers and, as Derrida (1982) suggests, one whose discourse is filled with hidden metaphors.

At the same time it is possible to see argument as a means of gaining knowledge as well as power about the world. Some would say that the two notions are always interlinked in this way. Argument as a means of clearing away the spurious, of asserting the real. The metaphors of building and journey are important here. A good argument is one that stands up to the onslaughts of time and tide; like a good theory. A journey is often an exploration of what is the case, a coming closer to the truth of the world and of the journeyer; hence our interest in travel books. An argument can also figure as part of a spiritual journey. Lakoff and Johnson ask us to imagine (ibid 5) a,

... culture where an argument is viewed as a dance, the participants are seen as performers, and the goal is to perform in a balanced and aesthetically pleasing way. In such a culture, people would view arguments differently, experience them differently, carry them out differently, and talk about them differently. But *we* would probably not view them as arguing at all: they would simply be doing something different.

I don't know of a culture where argument is viewed as a kind of dance but I do know of one where argument is viewed as part of the intellectual and spiritual path. In Buddhism (especially Tibetan Buddhism) young lamas perform a kind of ritualized argument. Not only are the steps highly formalized but the actual physical movements of the arguer are important. Characteristically the arguer punctuates or underlines a point by stretching out one hand towards the audience and striking it with the other. The whole thing is conducted in an atmosphere of great enjoyment. This seems to me to be a case where argument partakes of spirituality, knowledge, combat, performance and game at the same time. As Waddell, writing in the 19th century says "This exercise is called expressing "the true and innermost essence (of the doctrine)"..., in which an endeavour is made to ascertain both the literal sense and the spirit of the the doctrine...' (Waddell, 1974: 184).

There are two points in this disquisition: (a) the notion of argument is inherently metaphorical and (b) this notion is culturally determined. Thus even if someone were to succeed in automating argument s/he would only be capable (as a representative of one culture) of automating *one* view of argument. More importantly, the possibility of argument over the merits of Defoe which ended with one of the participants 'lying unconscious on the floor'. The ultimate grounding for one's beliefs one might say.

such automation is cast in doubt because of the metaphoricity of the discourse and (if Lakoff and Johnson are right) of the process. It is not just that arguments are spoken about metaphorically, they are thought about and acted out in a metaphorical manner. Thus while it may be possible to capture in a schematic form the notion of argument (as Lakoff and Johnson do) a machine would not be capable of arguing simply by virtue of having access to this schema. There are at least four reasons for this: (a) There is more to acting as if X is Y (that is, *acting* metaphorically) than the following of a schema; (b) While the change from viewing a conversation as a chat to viewing it as an argument must involve some cues these seem to be largely tacit and probably non-formalizable; (c) It is possible but probably not adequate to see this move in terms of a change in the schema determining the behaviour, however it seems more likely that such schemas are *post hoc* rationalizations or descriptions of pieces of behaviour rather than necessary elements in the internal processing which accompanies the behaviour; (d) There is more to thinking metaphorically than moving from one schema to another. This must involve at least some propensity to *merge* the schemas for two things.

8.2.3 Argument is culturally relative

We have already touched upon this in discussing Lakoff and Johnson. Another source for the notion of cultural relativism is the work of the later Wittgenstein, in particular the *Philosophical Investigations*. In the *Investigations*, Wittgenstein introduces the notion of a language-game. Wittgenstein's views on the nature of the language-game are summarized by Kenny (1973:163).

Language-games, like games, need have no external goal; they can be autonomous activities... But the comparison of language to a game was not meant to suggest that language was a pastime, or something trivial: on the contrary, it was meant to bring out the connection between the speaking of language and non-linguistic activities. Indeed the speaking of language is part of a communal activity, a way of living in society which Wittgenstein calls a 'form of life'... It is through sharing in the playing of language-games that language is connected with our life...

Language-games include the famous builders' language-game, and others such as (Wittgenstein, 1953: I, 23):

Giving orders, and obeying them

...

Reporting an event

...

Forming and testing a hypothesis

...

Making up a story; and reading it...

Play-acting

...

Making jokes

...

Asking, thanking, cursing, greeting, praying.

As the inclusion of 'Forming and testing a hypothesis' suggests, Wittgenstein can be taken to suggest that the whole process of the production of reasons to support claims (justification or argument) is itself a language-game. This is the opinion of Cook in a recent article (1988). He suggest that Wittgenstein believes that 'Explanation, justification occurs only within language-games'. Cook supports this with a quote from Wittgenstein's *Philosophical Grammar* (p. 97): 'A *reason* can only be given *within* a language-game. The links of the chain of reasons come to an end, at the boundary of the game.' Perhaps the most obvious way to gloss this rather gnomic utterance is to suggest that there can be alternative logics, alternative notions of what counts as a reason and how it is related to claims. Perhaps indeed there arises the possibility of a form of life in which the game of supporting claims with reasons is not played at all (as in Lakoff and Johnson's dance world). We have already touched upon the notion of relativity to a particular domain in our discussion of argument style in chapter 4. The notion before us is much more radical. Whether Wittgenstein is to be taken in this way is irrelevant for our purposes. The fact is that several philosophers have taken him in this manner. Winch, for example, in his discussion of the Zande (1970: 93/4) says:

...the forms in which rationality expresses itself in the culture of a human society cannot be elucidated *simply* in terms of the logical coherence of the rules according to which activities are carried out in that society. For, as we have seen, there comes a point where we are not even in a position to determine what is and what is not coherent in such a context of rules, without raising questions about the point which following those rules has in the society.

There are, in fact, strong arguments against the notion of cultural relativism which Winch and others have adopted. For instance, it might be argued that if all views are

culturally relative then the notion of cultural relativism itself must be culturally relative and not therefore applicable to all societies. However, there is enough plausibility in the notion to give us pause. Enough anyway to suggest why the building of an arguer may be such a difficult task. Arguments will always be understood and responded to in relation to a given cultural context. This may not mean that there is a different logic at work but it may mean that the point in the argument will be missed unless that background is shared.

It is important to see that there is no single background which is shared by all human beings. Geertz (1983:76) in discussing common sense as a cultural system points out that,

If common sense is as much an interpretation of the immediacies of experience, a gloss on them, as are myth, painting, epistemology, or whatever, then it is, like them, historically constructed and, like them, subjected to historically defined standards of judgment.

In short, common sense, as well as grounding our arguments, can also be the object of argument. Thus arguments take place against diverse backgrounds which are forever shifting. Until we can deal with all the different common senses possible and all the possible shifts we cannot be said to have captured a notion of human arguing, only that of a small section, a culturally specific part. And that process would, if possible, be a long and arduous one. As Geertz concludes (1983: 92):

If one wants to demonstrate, or even ... to suggest, that common sense is a cultural system, that there is an ingenerate order to it capable of being empirically uncovered, and conceptually formulated, one cannot do so by cataloguing its content, which is widely heterogeneous, not only across societies but within them - ant-heap wisdom. One cannot do so, either, by sketching out some logical structure which it always takes, for there is none. And one cannot do so by summing up the substantive conclusions it always draws, for there are, too, none of these. One has to proceed instead by the peculiar detour of evoking its generally recognized tone and temper, the untraveled side road that leads through constructing metaphorical predicates... to remind people of what they already know.

Geertz concludes that this notion of common sense would be a 'plenary jolt' for philosophy, since it relies on an unexamined notion of common sense. If that is so for philosophy it is even more so for artificial intelligence which relies on an unthinking mish-mash of researchers' own "common sense" perceptions and second hand notions derived from a particular philosophical tradition.

8.2.4 Argument includes tacit elements.

Arguing is a skill. I have glibly spoken above about knowledge of how to argue and have presented some representations which are designed to emulate some elements of that skill. However, there will be many elements which must remain tacit; things which an arguer can be said to know but which no arguer can articulate. As Janik (1988:59/60) points out:

There is nothing mysterious or implausible about this inability to put what we know into words... It is an essential element in what we normally term experience that experienced persons have a facility for handling a wide variety of unforeseen situations. Just as it is not possible to describe an unforeseen situation, by definition, it should be reasonable to think that our ability to deal with the situation should not be taken to imply that we *must* be able to articulate the knowledge which enables us to do so.

It might be suggested that, even if the arguer is unable to enunciate the nature of his or her skill, the cognitive scientist can capture it in a formal system which, by manipulating symbols, emulates an arguer. However, as Dreyfus and Dreyfus argue (1986:35) expert behaviour does not involve the manipulation of symbols in this way. The expert exhibits 'involved skilled behavior based on an accumulation of concrete experiences and the unconscious recognition of new situations as similar to whole remembered ones.' A system which follows rules of the kind in question would be at best merely a competent arguer. If expertise is an acquired skill, argument is even more so. The simulation of arguing is possible as I have shown above. However, there are good reasons for thinking that the machine can only be a passable arguer. These centre around the machine's lack of the requisite background knowledge. Furthermore, there is nothing to suggest that the manipulations captured in the simulation bear any resemblance whatever to the processes which underlie the argument of a human arguer. Stanley Fish (1982) makes similar points about interpretation:

Interpreters are constrained by their tacit awareness of what is possible and not possible to do, what is and is not a reasonable thing to say, and what will and will not be heard as evidence in a given enterprise; and it is within those same constraints that they see and bring others to see the shape of the documents to whose interpretation they are committed.

8.2.5 Argument touches on value systems.

As the quote from *Huckleberry Finn* in the introductory chapter shows, we don't often argue in an entirely objective way. We become involved because our values are at stake. This is related to the notion of argument as war discussed above. It is difficult to know how to deal with values in computational terms. An argument is pursued because some value is being threatened. Some value permeates an argument; guides the moves made in it. Without the value it would be impossible to understand what the point in the argument was. For example the debate about freedom of choice and whether we have it under this present, Thatcherite, government, depends on the prior understanding that freedom of choice is a good thing. Of course this could be introduced as a step in the argument. However there is more to a value than the assertion of a proposition. I may hold that it is an evil to kill another human being. I may also value human life. This latter means that I *act* towards human beings in certain ways. I do certain things in certain circumstances and avoid doing other things in other circumstances. The value permeates my whole way of acting and thinking. We humans, can understand and react to an argument because we understand and accept and value the same things. An ASP provides a means of integrating the proposition that it is wrong to kill humans into its knowledge base. In subsequent arguments this proposition will be used either for or against other propositions. However, the system does not act towards humans in some new way. It remains an open question as to whether a system can come to have a value as opposed to the acquisition of a belief which expresses a value. Furthermore, since values and emotions are tightly interconnected and emotions themselves are not beliefs or propositions, a computational arguer will never understand certain arguments.

8.2.6 Arguing is not something done in a cold, unemotional manner.

Schank (1979) touches upon the point I wish to raise. He puts forward the notion that one means of controlling inferences is by having a *interest value* which can be attached to concepts. The rules which will be fired are those with the highest interest rating. I am not really concerned with the application of the notion but only in the items which Schank counts as interesting. These are death, unexpected events, danger and personal relatedness. The point is that the fundamental grounding for our arguments (especially moral ones) is frequently in terms of human good and harm. If

we think of death and so on as the ultimate grounds of arguments, then, because there are so few of these it might seem possible to deal with them computationally with relative ease. Indeed there is work on the role of affect in the understanding of stories which suggests that the computational representation of emotions and values is possible (Hidi and Baird, 1986, Dyer, 1983). However I would argue that these concepts are knit into the nexus of emotions, values and instincts which is peculiar to us as living, experiencing animals. No machine can hope to share in this life. Thus no machine can have the interest in certain arguments. Of course a value could be set, as Schank suggests, and inferences made when certain concepts are over a certain threshold. But this is hardly the same thing as the emotions which grip us; which drive us on to pursue an argument (or indeed a train of thought). An emotion is not a proposition nor is a value. Often of course this emotion is negative and derives from our need to protect our egos from some perceived threat. Often however the emotion has positive outcomes. The emotion engendered by the interest drives the arguer to a positive outcome. Thus Crick and Watson, driven by personal ambition as well as scientific curiosity, achieved something of lasting significance. It seems to me that it is wrong to divorce arguing from emotions and interests. At the very least we take great pleasure in arguing. In this respect we do not live up to the ideal which Mr Spock sets himself; nor should we. Thus a machine to whom nothing can matter cannot be a real arguer (Edelson, 1986).

8.2.7 Arguments are never divorced from personal, social and political considerations. There is a political dimension to arguments as there is to much else. Debate is often contrasted with confrontation as a means of settling disputes. Thus we have arbitration for industrial disputes and the United Nations as mediator in war situations. As Churchill said 'Jaw, Jaw is better than War, War'. Argumentation, through the medium of books or journalism is often seen as a means of getting things changed. As the process by which opposing views are revealed, debated and criticised. At the centre of our democracy is the debating chamber of the House of Commons. However it is impossible to resist commenting that while free speech is regarded as a right in a democracy, all states, including Western democracies, will silence debate when the state feels threatened in some manner. This is particularly true of the present government which, on a range of issues ranging from the Zircon affair to the Peter Wright book *Spycatcher*, has shown an unwillingness to allow free

debate. As Enoch Powell said in the Guardian of 3-8-87, 'In real life the strength of any case is not the strength of any arguments for it, but the strength of the political, personal and often economic investments which have been made in it.' We cannot hope to have much free debate in a society in which the press is owned by a few millionaires, television is attacked for the slightest anti-government opinion and parliament itself is simply a means of expressing overwhelming right wing power. At the same time, given that the only alternative is the gun, we must encourage debate and teach skills in debate so that individuals can assess and oppose the arguments of those with real power in our society. Argument (and artistic creations) are the only means of achieving this. As Fisher (1988:1) says,

We learn most of what we know from teachers and experts of one kind and another and this is not surprising in a highly specialised modern society. However, it is possible to rely too heavily on experts and this approach to learning and knowledge tends to encourage passivity and receptiveness rather than inventiveness and imagination.

Argument is one means of combating the relative impotence of the individual in modern industrialised society.

Argument is politically and socially determined in another way. As a result of political changes a shift can occur in a culture so that where there were once clear notions, say, in the moral sphere, rooted in fixed patterns of political and social control, the meanings of the terms become loose, multivalued and fuzzy. This notion derives from Newbold's (1988: 229) discussion of the contrast between polysemy and univocality.

A univocal code can be used to construct complex chains or hierarchies of reasoning, since every relatively unambiguous step in the process can be grasped. Univocality and the precise definition of terms is necessary for abstract argument, the linear progression of syllogism, jurisprudence, and the efficient functioning of organizations such as large armies and empires. Polysemy undermines pyramidal and convergent structures, and favors imagination, intuition, unpredictable leaps, fields of association, and lateral, fluid relationships.

Newbold is discussing the breakdown in authority in the late Roman empire and the resulting changes in political discourse. Since there are close analogies between the state of late Roman civilization and the contemporary west it may well be that we are in a period of transition to a polysemous code.

8.2.8 Arguments are creative.

We hold as obvious both the view that reason is mechanical and that reason is mysterious. The mechanical view envisages thinking as the transparent cranking of an old hand-cranking machine... This is the view of reason ... as a set of operations which can be mechanically performed. I do not understand the psychology of it, why it seems comforting to view reason as a transparent machine and why the view of the mechanical or automatic part of reasoning seems clearer than the logician's view of reason as deductive validity of inferences, no matter how they were arrived at. This logician's view seems somehow disturbing... it is much inferior, emotionally, to the view that there is something mysterious and magical in the operations of reason. The logician's view... may be supplemented by the view that the premises of rational thinking are arrived at neither by calculation nor by a mysterious and magical act, but by a mysterious nonmagical act, or by a nonmysterious, nonmechanical act, or by no act at all but by sheer luck. (Agassi, 1982:105)

We can apply what Agassi says about reason to argument and debate. Even if we can use mechanical (computational procedures) in the justification of a piece of reasoning or an interpretation (the context of justification) we cannot, of necessity, apply these in the creation of the interpretation, the finding of a novel argument in the first place (the context of discovery). This will always be essentially mysterious. The creation of an argument like the creation of poem or a novel, involves a complex range of human attributes. Conrad knew this well as the following quote from *Under Western Eyes* suggests:

Some superior power had inspired him with a flow of masterly argument as certain converted sinners become overwhelmingly loquacious.

Milton, in common with many great poets requests the aid of a superior power in creating his 'great argument' - the text of *Paradise Lost*:

what in me is dark
Illumine, what is low raise and support,
That, to the height of this great argument,
I may assert Eternal Providence,
And justify the ways of God to men.

We need to know the domain to be able to argue relevantly, we must also have some idea of the kinds of inference which are permitted in the domain, the shared background, the current context of debate and so on. But arguments are rarely simple functions of all these elements. Good arguments are often novel. New aspects of the domain are noticed and expressed, new ways of relating elements of the domain are

created. Whatever view we take of creativity we must admit that arguments are creative. We might think (as Weisberg does, 1986) that creativity is a form of problem-solving in which we recognize similarities between the present case and a previous one and apply the lessons learned from the previous case (suitably modified) to the present case. This process is more or less the same as that of analogical problem solving mentioned above. Creativity, in this view, is not something mysterious. On the other hand we might have a view of creativity in which (as Poincaré thought) we first gather knowledge about a problem and then allow a period of sub-conscious activity during which a solution appears. The real work is then in checking that the idea works. Whatever the notion of creativity we hold we must admit that an argument is more than the production of some conclusion by means of a series of rewrite rules. If Weisberg is correct, argument must at least involve analogical problem-solving. This in turn depends on a computational theory of analogical problem solving. There have been many attempts at this (see references in section 4.3 above) but it is not clear that any of them is fully successful. If Poincaré is right, and the process is hidden and hence mysterious, it seems that since no formal rules can be extracted no computational model can be made. I think that the truth is somewhere in between. There is more to the production of a good argument than analogical transfer. How, for instance do we hit upon the right analogy, is it purely serendipitous? At the same time there is nothing mysterious about the creation of good arguments. We can teach students how to come up with ideas and how to marshal them subsequently (see Pirie 1985 for a good example). This of course does not mean that we can capture this process in a formal manner. At best this must remain an open question. From my own experience of producing arguments, the process is not guided by any formalizable rules. It is iterative and messy. Good arguments arise as a result of interactions with others (as the Russian director, Tarkovsky, suggests in the epigraph to this thesis), reading, building partial models (as Crick and Watson did). The formation of arguments (the context of discovery) includes a lot of processing which takes place before the argument can be clearly enough stated to be assessed in the manner described in the body of this thesis. There is a sort of negative capability at work here. This notion comes from the poet Keats as explained by Ward (1966:160/1):

"Several things dovetailed in my mind," Keats wrote his brothers, "& at once it struck me, what quality went to form a Man of Achievement especially in Literature & which Shakespeare possessed so enormously - I mean *Negative*

Capability, that is when man is capable of being in uncertainties, Mysteries, doubts, without any irritability reaching after fact & reason." This ability to "remain content with half knowledge" - what today is called "tolerance for ambiguity" - was, as Keats saw it, essential to the poet insofar as he above all men explores the frontiers of human experience and struggles with its endless diversity and contradictions in an effort to extend the limits of human awareness.

We are capable of floundering around with a half-baked idea and still come up with something useful. All great detectives work like this. Can a machine? It is here that the notion of the background, the affective elements, the culture and the value system play a part. An arguer must understand and feel a great deal before s/he can produce a new argument from half-baked ideas.

8.2.9 Interpretation is not simply the application of transformations to data.

For instance Shanks and Tilley (1987) suggest that the process of interpretation is fourfold. They refer to it as the 'fourfold hermeneutic'. It is interesting and perhaps not coincidental that Shanks and Tilley, in reacting against what they see as naive empiricist approaches to archaeology, adopt elements from the same tradition as is used by Winograd and Flores (1986) in their attack on the rationalism of artificial intelligence research. They describe the fourfold hermeneutic as follows (ibid: 108):

- (i) the hermeneutic of working within the contemporary discipline of archaeology;
- (ii) the hermeneutic of living within contemporary society as an active participant, put broadly, gaining knowledge of that which is to be human, to interact and participate with others and to be involved in struggles about beliefs and social and political values;
- (iii) the hermeneutic of trying to understand an alien culture involving meaning frames radically different from our own;
- (iv) the hermeneutic involved in transcending past and present.

Perhaps (iii) and (iv) have no relevance to AI as a discipline but (i) and (ii) do. New hermeneutics (iiia) and (iva) could be coined:

- (iiia) the hermeneutic of trying to understand the mechanisms which produce the behaviour which is interpreted by means of the other hermeneutics.
- (iva) the hermeneutic involved in integrating people and machines.

At any rate there is more to interpretation than we have discussed. We have gone

some way to dealing the common sense and other theories which ground interpretations and hence arguments. We have even attempted (in the notion of argument style) to deal with differences between disciplines. We have done nothing to deal with the economic and political forces which feed an interpretation or with the 'alien meaning frames' within which the objects interpreted had a place.

The upshot of this (largely negative) discussion is that we must be cautious in ascribing the ability to argue to a machine. Although we have gone some way towards understanding what it is to argue and to capturing this in a model sufficiently precise for a computer program to be designed and implemented, we cannot truthfully be said to have simulated an arguer. There are many outstanding problems in AI which would need to be solved first. Intelligence is not composed of easily separable components. The ability to argue cannot be simulated without the prior ability to simulate many diverse human abilities. As the novelist John Barth says in his novel, *The Floating Opera*, 'I think that to understand any one thing entirely, no matter how minute, requires the understanding of every other thing in the world'. This is not a counsel of despair. The program may not really argue but it provides a valuable service. And, as we shall see in the next section, we can make use of the knowledge gained to build viable and stimulating programs for the dissemination of knowledge.

8.3 Future directions

I have already discussed the particular extensions and amendments which need to be made to ASParch if it is to function as a usable tool for archaeologists and others. In this section I will discuss some ideas for incorporating this research into further work.

One possible direction for future work would be the integration of work done on KIVA (the archaeological interpreter mentioned in chapter 5) and ASParch. This hybrid system (illustrated in Figure 8.1) would form part of a toolbox (or archaeologist's workbench) which would aid in the creation of knowledge bases, their interpretation and interaction with them. I have already begun to think about the design for this system which would be known as WORSAAE (after the great Danish archaeologist). This system has the potential to provide a means of controlling the interaction between users and knowledge bases in terms of a model of stylized argument exchange which renders the knowledge base contents accessible, justifiable

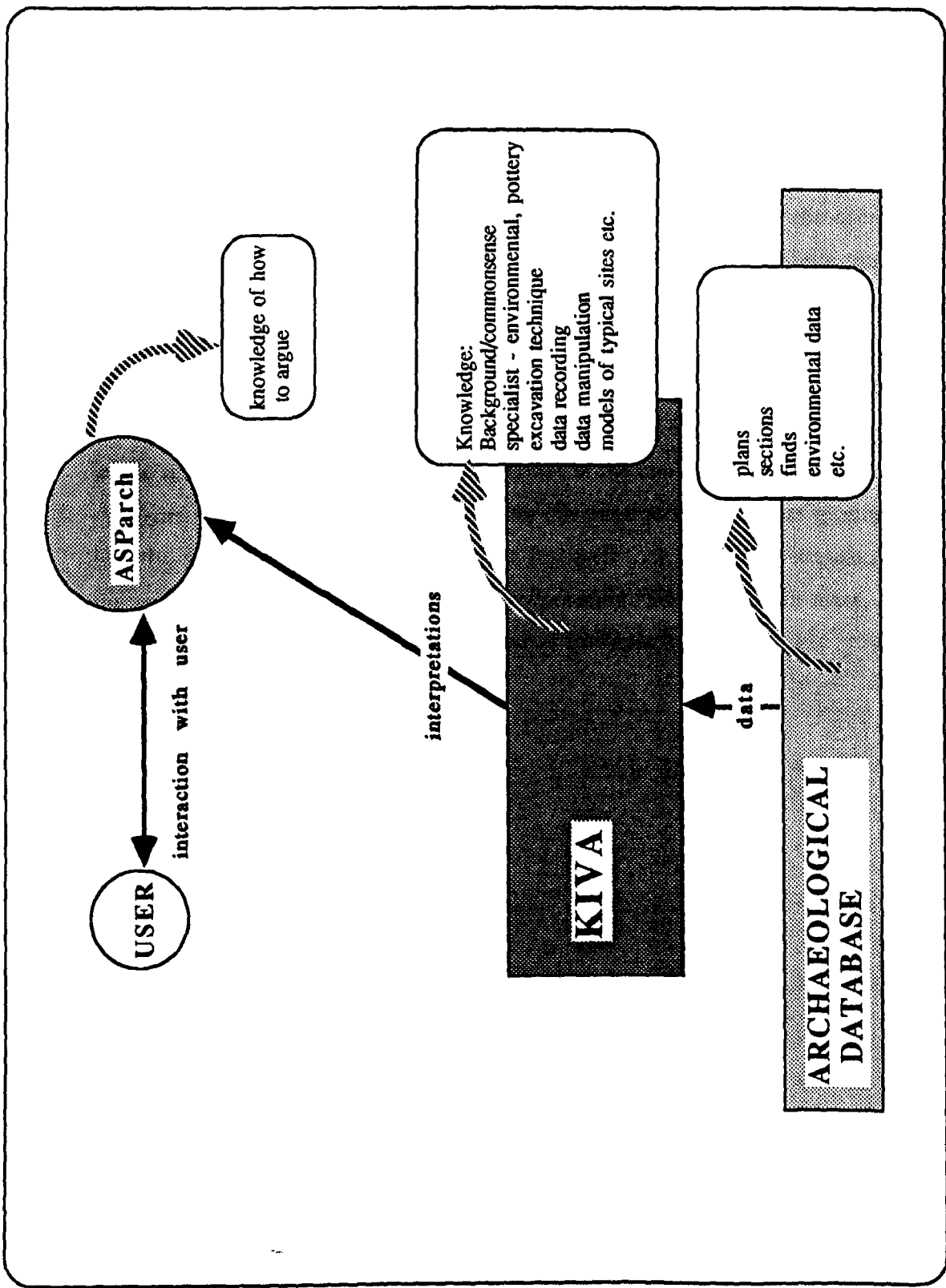


Figure 8.1 - The hybrid system

and changeable. As a tool, such a system, in modelling an arguing agent as a means of providing the interface between the user and archaeological knowledge stored in an electronic form, would provide reasoned positions about the contents of its knowledge store and allow the user to change and update this knowledge. Such a system is also ideally suited to a role in education/dissemination of knowledge because it includes an argumentation module. It is my belief that what is lacking in the notion of an expert system as a communicator of knowledge is some means of involving the user. Our personal experience shows us that argumentation, because it must by its nature be geared to the specific interests and points of view enunciated by the user, is an important means of achieving this (cf. Schwartz 1988). We also hold that a great deal of what is important about any domain is captured in the arguments surrounding the data rather than the data. As Lowe (1985: 98) suggests in discussing his system for mediating the arguments of multiple users, the 'goal of this medium is to combine the *content* of many contributions on any given topic into a single structure'. That is to imply that knowledge is not held in facts but in the chains of argumentation which link the facts to each other and to the various interpretive theories that ground them. Finally, such a system can perform as a research tool in that the archaeologist can use the system to test arguments before incorporating them in written texts. The possible uses for WORSAAE can be summarized as in Table 8.1.

Area	Use
expert systems, tutoring systems	WORSAAE as an interface which can provide chains of argumentation
paper publication - i.e. conventional printed or <i>samizdat</i>	WORSAAE as a pre-processor of data aimed at producing rationales for claims
machine readable publication - i.e. where machines have access to <i>full</i> data and can autonomously derive the required information	WORSAAE as a search mechanism for evidence pro and contra a particular position.

Table 8.1 - Possible uses for the successor to ASParch

The last of these is the most speculative. Here the notion is that the system has access to all the data from an excavation and can derive an explanation/interpretation and correlate it with current beliefs. This data need not be published in any paper form (cf the notion of electronic periodicals such as BLEND - Shackel, 1982.). Thus the archaeologist could set WORSAAE off in a search through a given data-base (or set of databases) in an attempt to find supporting or conflicting evidence. This could then be incorporated into his or her own work. In this way, the paper publication of the data is limited to particular interests. While there has been some work on the the scanning of texts for arguments and on monitoring output from news agency lines (Alvarado et al., 1986 on argument in newspaper editorials, Dejong, 1979 and Lebowitz, 1980, on the monitoring of news agency lines), nonetheless this probably represents a very long-term goal.

A related possibility also suggests itself. The above shows how ASParch could function as part of a tool box for aiding academic research. This has already been suggested for other areas (for instance, Marc Eisenstadt, in a lecture at the Open University which is unpublished, suggests that a system which could argue back would be helpful in the formation of theories in the domain of neuroscience). The

second possibility is an extension of the system's role as part of the interface between the user and the knowledge or database. In pursuing this role it should be possible for ASParch to act as an intelligent (and argumentative) guide through a hypertext system.

Halasz (1988) has the following comment about AI and Hypertext (mentioned already in chapter 3):

The integration of hypermedia and AI technology is an interesting direction to explore. In many ways, hypermedia and knowledge-based systems are a natural fit... A merging of concepts from frame-based systems into the design of hypermedia systems would be a sensible way to approach the integration of hypermedia with rule-based, truth-maintenance, and other computational engines.

While such an integration of hypermedia and artificial intelligence is a worthwhile goal, such an integration is best achieved not at the implementational level suggested in the above quote but in terms of a model of a higher level cognitive process which combines the discursiveness of hypertext with the inferential capabilities of expert systems and which is at the same time an essential aspect of research in the humanities. I suggest that a computational model of argument fits this description. Inference is necessary for argument assessment and for selecting an appropriate response. Hypertext is necessary for the storing and display of an argument. The resources of a hypertext document could also be used for exemplification and illustration. The resulting system is one which has much in common with human/human arguments but which differs in significant ways. The machine is capable of argument assessment and response and thus provides a natural means of interaction. At the same time the system goes beyond human capabilities. It can display several premises for an argument at the same time in different windows. It can display all the links between claims and grounds, supporting and attacking arguments and provide the means for navigating around these links.

The contribution of Trigg and Weiser (1986) and others on the use of hypertexts as tools for argumentation (discussed above) shows how integration of the argumentation model and hypermedia is possible. These systems provide a means of *storing* and *accessing* rather than *assessing* or *responding* to arguments. They also provide a means of storing large amounts of diverse domain knowledge.

The system I envisage incorporates many of the features associated with other

hypertext systems. An argument will proceed by means of links from one chunk of argument text to another text. These links will include comment, criticism, redescription and so on. At the same time the system will be capable of *interacting* with the user. It will be able to *understand* user arguments and, as part of the comprehension process, *check* argument steps and overall arguments. It will also be able to *produce* arguments and keep an argument going.

This system thus integrates AI and hypertext not in terms of a low level representational formalisms or techniques but in terms of a human-human interaction which can be captured on a machine and can naturally be seen to make use of AI and hypertext. From the hypertextual point of view, the system would act as a kind of intelligent but argumentative guide through complex and diverse domain knowledge. From the AI point of view, the argumentation module would be enhanced by the flexible integration of diverse sorts of data possible in a hypertext system as well as by the capacity to thread the various data together into manifold chains of argumentation which the user can explore at leisure.

Epilogue

We started with a reference to Rousseau's ideas about political freedom. We end with the Buddha's notion that error will continue until we become detached from the blindness of everyday life so that we can see the real; Nirvana. There is a famous Buddhist parable which provides a rationale for the attempt to understand our need and capacity to argue. Here it is suggested that we are all possessed of part of the truth but convinced that we are privy to it all. While mere quarrels are looked upon with disdain, the Buddha, like Plato, seems to have seen intellectual debate as a spiritual endeavour: selfless but not without compassion; detached but not coldly abstract; *anatta* (the Buddhist doctrine of no-soul) but not nihilistic. Debate is of vital importance to our spiritual health and to that of our democracies. As the necessary concomitant of toleration it is needed more and more in this pluralistic world we live in.

Then said the Exalted One:

'These sectarian, brethren, are blind and unseeing. They do not know the real, they know not the unreal, know not the truth, know not the untruth: in such a state of ignorance do they dispute and quarrel as ye describe. Now in former times,

brethren, there was a rajah in this same Savatthi. Then, brethren, that rajah called to a certain man, saying: "Come thou, good fellow! Go and gather together all the blind men that are in Savatthi!"

"Very good, your majesty," replied that man, and in obedience to the rajah gathered together all the blind men, took them with him to the rajah and said:

"Your majesty, all the blind men of Savatthi are now assembled."

"Then, my good man, show these blind men an elephant."

"Very good, your majesty," said the man, and did as he was told, saying, "O ye blind, such as this is an elephant!"

And to one man he presented the head of the elephant, to another the ear, to another the tusk, the trunk, the foot, back, tail, and tuft of the tail, saying to each one that that was the elephant.

Now, brethren, that man having presented the elephant to the blind men, came to the rajah and said, "Your majesty, the elephant has been presented to the blind men. Do what is your will."

Thereupon, brethren, that rajah went up to the blind men and said to each, "Have you studied the elephant?"

"Yes, your majesty."

"Then tell me your conclusions about him."

Thereupon those who had been presented with the head answered, "Your majesty, an elephant is just like a pot." And those who had only observed the ear replied, "An elephant is just like a winnowing-basket." Those who had been presented with the tusk said it was a ploughshare. Those who knew only the trunk said it was a plough. "The body," said they, is a granary: the foot, a pillar: the back, a mortar: the tail, a pestle: the tuft of the tail, just a besom." Then they began to quarrel, shouting, "Yes it is! No it isn't! An elephant is not that! Yes, it's like that!" and so on, till they came to fisticuffs about the matter.

Then, brethren, that rajah was delighted with the scene.

Just so are the sectarians, who are wanderers, blind, unseeing, knowing not the truth, but each maintaining it is thus and thus.²

-o0o-

² From Woodward, 1973. The parable is from the *Udana*, vi, 4.

References

-oOo-

- Abelson, R.P. The structure of belief systems. In: Schank, R.C. and Colby, K.M. eds. *Computer Models of Thought and Language*. San Francisco: W.H. Freeman and Co., 1973.
- Adam, A.E. and Taylor, A.D. Modelling analogical reasoning for legal applications. In: *Research and Development in Expert Systems III*. ed. M.A. Bramer (Expert Systems-86). Cambridge: Cambridge University Press, 1987.
- Agassi, J. The detective novel and scientific method. *Poetics Today*, 3:1, 99-108, 1982.
- Alvarado, S.J., Dyer, M.G. and Flowers, M. Memory representation and retrieval for editorial comprehension. *Proc. Seventh Annual Conf. of the Cognitive Science Society*, 228-235, 1985.
- Alvarado, S.J., Dyer, M.G. and Flowers, M. Editorial comprehension in OpEd through argument units. *Proceedings of the Fifth National Conference on Artificial Intelligence (AAAI-86)*, Philadelphia, PA, 250-256, 1986.
- Anderson, A.R. and Belnap, N.D. *Entailment*, Vol 1. Princeton: Princeton University Press, 1975.
- Anderson, K.M. Ethnographic analogy and archaeological interpretation. *Science*, vol 163, no 3863, 133-138, January 1969.
- Baker, K.G. Archaeology and expert systems: some problems encountered during practical work. *Proceedings of the Third International Expert Systems Conference*, London, 211-219, 1986.
- Baker, M., Burstein, M.H. and Collins, A.M. Implementing a model of human plausible reasoning. *Proceedings of the Tenth International Joint Conference on Artificial Intelligence (IJCAI-87)*, 185-188, Los Angeles: Morgan Kaufmann, 1987.
- Baker, M. Arguing with the tutor: a model for a tutorial dialogue in uncertain knowledge domains. CITE Report No. 51, Institute of Educational Technology, The Open University, 1988.
- Begeman, M.L. and Conklin, J. The right tool for the job. *Byte*, 255-266, October 1988.
- Binford, L.R. Smudge pits and hide smoking: the use of analogy in archaeological reasoning. *American Antiquity*, 32, 1-12, 1967.

- Binford, L.R. *In Pursuit of the Past: decoding the archaeological record*. London: Thames and Hudson, 1983.
- Binford, L.R. Researching ambiguity: frames of reference and site structure. In: ed. Kent, S. *Method and Theory for Activity Area Research*. New York: Columbia University Press, 1987.
- Birnbaum, L. Argument molecules: a functional representation of argument structure. *Proceedings of the Second National Conference on Artificial Intelligence (AAAI-82)*, 1982.
- Bishop, M.C. and Thomas, J. "BEAKER" - An expert system for the BBC Micro. *Computer Applications in Archaeology*, 56-62, 1984.
- Bonnichsen, R. Millie's Camp: an experiment in archaeology. *World Archaeology*, 4(3), 277-291, 1973.
- Boose, J.H. A knowledge acquisition program for expert systems based on personal construct psychology. *International Journal of Man-Machine Studies*, 23, 495-525, 1985.
- Boose, J.H. and Bradshaw, J.M. Expertise transfer and complex problems: using AQUINAS as a knowledge-acquisition workbench for knowledge-based systems. *International Journal of Man-Machine Studies*, 26, 3-28, 1987.
- Branigan, K. *The Catuvellauni*. Gloucester: Alan Sutton Publishing Ltd, 1985.
- Bratko, I. *Prolog programming for artificial intelligence*. Wokingham: Addison-Wesley, 1986.
- Brough, D.R. and Parfitt, N. An expert system for the ageing of a domestic animal. *Conference on Computer Applications in Archaeology*, 1984.
- Brough, D.R. and Alexander, I.F. The Fossil expert system. *Expert Systems*, 3(2), 76-83, 1986.
- Buchanan, B. and Shortliffe, E.H. *Rule-Based Expert Systems: The Mycin experiments of the Stanford Heuristic Programming Project*. Reading, Mass: Addison-Wesley, 1984.
- Burrows, J.F. *Computation into Criticism*. Oxford: Clarendon Press, 1987.
- Bylander, T. and Chandrasekaran, B. Generic tasks for knowledge-based reasoning: the "right" level of abstraction for knowledge acquisition. *International Journal of Man-Machine Studies*, 26, 231-243, 1987.

- Calvino, I. *The Uses of Literature*. San Diego: Harcourt Brace Jovanovich Publishers. 1986.
- Carbonell, J. Politics. In: Schank, R.C. and Riesbeck, C.K. eds. *Inside Computer Understanding: Five programs plus miniatures*. Hillsdale, New Jersey: Lawrence Erlbaum Assoc., 1981.
- Case, H. Neolithic explanations. *Antiquity*. 43, 176-186, 1969.
- Catano, J.V. Poetry and computers: experimenting with the communal text. *Computers and the Humanities*. 13, 269-275, 1979.
- Cheng, P.W. and Holyoak, K.J. Pragmatic reasoning schemas. *Cognitive Psychology*, 17, 391-416, 1985.
- Cheng, P.W., Holyoak, K.J., Nisbett, R.E. and Oliver, L.M. Pragmatic versus syntactic approaches to training deductive reasoning. *Cognitive Psychology*, 18, 293-328, 1986.
- Chouraqui, E. Construction of a model for reasoning by analogy. In: *Progress in Artificial Intelligence*. eds: Steels, L. and Campbell, J.A. Chichester: Ellis Horwood Ltd, 1985.
- Clancey, W.J. Tutoring rules for guiding a case method dialogue. *Int. Journal of Man-Machine Studies*, 11, 25-49, 1979.
- Clancey, W.J. The epistemology of a rule-based expert system: a framework for explanation. *Artificial Intelligence*, 20, 215-251, 1983.
- Clancey, W.J. Acquiring, representing and evaluating a competence model of diagnostic strategy. Knowledge Systems Lab, Stanford. Report No. KSL-84-2, August, 1985a.
- Clancey, W.J. Heuristic classification. *Artificial Intelligence*, 27, 289-350, 1985b.
- Clancey, W.J. From Guidon to Heracles in twenty short lessons: ORN Final Report 1979-1985. *AI Magazine*, August, 1986a.
- Clancey, W.J. The science and engineering of qualitative models. Working Paper No. 86-27, Knowledge Systems Laboratory, Stanford, 1986b.
- Clancey, W.J. and Letsinger, R. Neomycin: reconfiguring a rule-based expert system for application to teaching. *Proceedings of the Seventh International Joint Conference on Artificial Intelligence (IJCAI-81)*, 1981.
- Clarke, D.V., Cowie, T.G. and Foxon, A. *Symbols of Power at the time of Stonehenge*,

- HMSO/National Museum of Antiquities of Scotland, Edinburgh, 1985.
- Clarke, G. The invasion hypothesis in British archaeology. *Antiquity*, XL, 1966.
- Cohen, P.R. *Heuristic Reasoning about Uncertainty: An Artificial Intelligence Approach*. London: Pitman, 1985.
- Cohen, P.R. and Grinberg, M.R. A framework for heuristic reasoning about uncertainty. *Proceedings of the Eighth International Joint Conference on Artificial Intelligence (IJCAI-83)*, 1983.
- Cohen, R. Analyzing the structure of argumentative discourse. *Computational Linguistics*. 13(1-2), 11-24, 1987.
- Coles, J. *Archaeology by Experiment*. London: Hutchinson & Co, 1973.
- Collins, A. Fragments of a theory of human plausible reasoning. *Proceedings of the conference on Theoretical Issues in Natural Language Processing-2 (TINLAP-2)*, 194-201, New York: ACM, 1978.
- Collins, A. and Michalski, R. The logic of plausible reasoning: a core theory. To appear in *Cognitive Science*.
- Conklin, J. Hypertext: an introduction and survey. *IEEE Computer*, 17-41, September 1987a.
- Conklin, J. A survey of hypertext. MCC Technical Report No. STP-356-86, Rev. 2. MCC: Austin, Texas, December 1987b.
- Cook, J.W. Wittgenstein and religious belief. *Philosophy*, 63(246), 427-452, 1988.
- Coombs, M. and Alty, J., Expert systems: an alternative paradigm. *International Journal of Man-Machine Studies*, 20, 21-43, 1984.
- Corns, T.N. and Smith, M.E. Literature. In: *Information Technology in the Humanities*. ed. Rahtz, S. Chichester: Ellis Horwood, 1987.
- Culler, J. Interpretations: data or goals? *Poetics Today*, 9:2, 275-290, 1988.
- Curren, C.B. Potential interpretations of "stone gorget" function. *American Antiquity*. 42, 97-101, 1977.
- D'Andrade, R.G. The cultural part of cognition. *Cognitive Science*, 5, 179-195, 1981.
- Darwin, C. *The Illustrated Origin of Species*. London: Faber and Faber, 1979.

- David, N, Sterner, J. and Gavua, K. Why Pots Are Decorated. *Current Anthropology*, 29(3), 365-389, 1988.
- Davies, T.R. and Russell, S.J. A logical approach to reasoning by analogy. *Proceedings of the Tenth International Joint Conference on Artificial Intelligence (IJCAI-87)*, 264-270, Los Angeles: Morgan Kaufmann, 1987.
- Davis, R. Meta-rules: reasoning about control. *Artificial Intelligence*, 15, 179-222, 1980.
- DeJong II, G.F. *Skimming Stories in Real Time: An Experiment in Integrated Understanding*. Technical Report 158, Dept. of Computer Science, Yale University, 1979.
- de Kleer, J. AI approaches to troubleshooting. In: *Artificial Intelligence in Maintenance*. ed. Richardson, J.J. New Jersey: Noyes Publications, 1986.
- De Millo, R.A., Lipton, R.J. and Perlis, A.J. Social processes and proofs of theorems and programs. *Communications of the ACM*, 22(5), 271-280, 1979.
- Derrida, J. White mythology: Metaphor in the text of philosophy. In: *Margins of Philosophy*. Chicago: University of Chicago Press, 1982.
- Doran, J. Knowledge representation for archaeological inference. eds. Elcock, E.W. and Michie, D. *Machine Intelligence 8*. Edinburgh: Edinburgh University Press, 1977.
- Doran, J. A computational model of sociocultural systems and their dynamics. In: eds. Renfrew, C., Rowlands, M.J. and Segraves, B.A. *Theory and Explanation in Archaeology*. New York: Academic Press, 1982.
- Doran, J. A contract-structure model of sociocultural change. Conference on Computer Applications in Archaeology, 1986.
- Doran, J. The structure and emergence of hierarchical organisations. Alvey Workshop on Multiple Agent Systems, Redhill, April 14-15, 1988.
- Doyle, J. A truth maintenance system. *Artificial Intelligence*, 12, 231-272, 1979.
- Doyle, J., *A model for deliberation, action and introspection*. Tech. Report AI TR-581, MIT AI Laboratory, 1980.
- Draper, S.W. A user-centred concept of explanation. *Alvey KBS Club Second Workshop on Explanation*, January 1987.
- Dreyfus, H.L. and Dreyfus, S.E. *Mind Over Machine*. Oxford: Basil Blackwell Ltd, 1986.
- Dworkin, R. Law as Interpretation. *Critical Inquiry*, 9, 179-200, September 1982.

- Dyer, M. The role of affect in narratives. *Cognitive Science*, 7, 211-242, 1983.
- Eco, U. *Reflections on the Name of the Rose*. London: Secker and Warburg, 1985.
- Eco, U. and Sebeok, T.A. *The Sign of Three: Dupin, Holmes, Peirce*. Bloomington: Indiana University Press, 1983.
- Edelson, T. Can a system be intelligent if it never gives a damn? *Proceedings of the Fifth National Conference on Artificial Intelligence (AAAI-86)*, 1986.
- Eisenstadt, M. and Brayshaw, M. The Transparent Prolog Machine (TPM): An execution model and graphical debugger for logic programming. Technical Report No. 21a, (To appear in *Logic Programming*), HCRL, The Open University, 1987.
- Ennals, J.R. *Artificial Intelligence: Applications to Logical reasoning and Historical Research*. Chichester: Ellis Horwood, 1985.
- Ennals, J.R. *Star Wars: A question of Initiative*. Chichester: John Wiley and Sons, 1986.
- Ennals, J.R. and Brough, D.R. Representing the knowledge of the expert archaeologist. *Computer Applications in Archaeology*, 1982, pp. 133-44.
- Erman, L.D., Hayes-Roth, F., Lesser, V.R. and Reddy, D.R. The Hearsay-II Speech-Understanding system: Integrating Knowledge to Resolve Uncertainty. *Computing Surveys*, 12(2), 213-253, 1980.
- Eshelman, L., Ehret, D., McDermott, J. and Tan, M. MOLE: a tenacious knowledge-acquisition tool. *International Journal of Man-Machine Studies*, 26, 41-54, 1987.
- Fink, P.A., Lusth, J.C. and Duran, J.W. A general expert system design for diagnostic problem solving. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-7, 5, 553-560, 1985.
- Fish, S. Working on the chain gang: Interpretation in the law and in literary criticism. *Critical Inquiry*. 9. Sept 1982.
- Fisher, A. *The logic of real arguments*. Cambridge: Cambridge University Press. 1988
- Flowers, M., McGuire, R. and Birnbaum, L. Adversary arguments and the logic of personal attacks. In: *Strategies for Natural Language Understanding* eds. Lehnert, W.G. and Ringle, M.G. Hillsdale, NJ: Lawrence Erlbaum Associates, 1982.
- Flowers, M. and Dyer, M.G. Really arguing with your computer in natural language. *Proceedings of the National Computer Conference*, 651-659, 1984.
- Fowler, R. *Linguistic Criticism*. Oxford: Oxford University Press, 1986.

- Francfort, H.-P. Modélisation de raisonnements interprétifs en archéologie à l'aide de systèmes-experts: conséquences d'une critique des fondements des inférences. Paper prepared for the 2nd Anglo-French Round Table "Exploring the Foundations of Interpretation in the Humanities: The Role of Artificial Intelligence", Lincolnshire, September 1988.
- Frere, S. *Britannia: A History of Roman Britain*. London: Routledge and Kegan Paul Ltd, 1967.
- Gammack, J.G. and Young, R.M. Psychological techniques for eliciting expert knowledge. In: *Research and Development in Expert Systems (ES84)* ed. Bramer, M. Cambridge: Cambridge University Press, 1985.
- Gardin, J-C. *Archaeological constructs*. Cambridge: Cambridge University Press, 1980.
- Gardin, J-C. et al. *La Logique du Plausible*. Paris: Editions de la Maison des Sciences de l'Homme, 1981.
- Gardin, J-C. et al. *Systèmes experts et sciences humaines: Le cas de l'archéologie*. Paris: Eyrolles, 1987a. (Translated by Ennals, J.R. as *Artificial Intelligence and Expert Systems: Case studies in the knowledge domain of archaeology*. Chichester: Ellis Horwood, 1988.)
- Gardin, J-C. The future influence of computers on the interplay between research and teaching in the humanities (a summary of a presentation to CATH-87). *Humanities Communication Newsletter*. 9, 1987b.
- Garfinkel, A. *Forms of Explanation*. Yale: Yale University Press, 1981.
- Geertz, C. *Local Knowledge*. New York, Basic Books Inc., 1983.
- Genesereth, M.R. and Smith, D.E. Meta-level architecture. *Memo HPP-81-6 Stanford Heuristic Programming Project*. Stanford: Dept. of Computer Science, Stanford University, 1982.
- Gentner, D. Structure-mapping: a theoretical framework for analogy. *Cognitive Science*, 7, 155-170, 1983.
- Goguen, J.A., Weiner, J.L. and Linde, C. Reasoning and natural explanation. *International Journal of Man-Machine Studies*, 19, 521-559, 1983.
- Gould, R.A. and Watson, P.J. A dialogue on the meaning and use of analogy in ethnoarchaeological reasoning. *Journal of Anthropological Archaeology*, 1, 355-381, 1982.

- Grice, P. Logic and conversation. In: eds. Cole, P. and Morgan, J. *Syntax and Semantics*. Academic Press, 1975.
- Halasz, F. G. Reflections on Notecards: Seven issues for the next generation of hypermedia systems. *Communications of the ACM*, Vol 31, No 7, 836-852, July 1988.
- Harman, G., *Change in View*. Cambridge, Mass.: MIT Press, 1986.
- Hasling, D.W., Clancey, W.J. and Rennels, G. Strategic explanations for a diagnostic consultation system. *International Journal of Man-Machine Studies*, 20, 3-19, 1984.
- Hempel, C.G. Explanation in science and in history. In: *The Philosophy of Science*, ed. Nidditch, P.H. Oxford: Oxford University Press, 1968.
- Hempel, C.G. Thoughts on the limitations of discovery by computer. In: Schaffner, K.F. ed. *Logic of Discovery and Diagnosis in Medicine*. Berkeley: University of California Press, 1985.
- Henle, M. On the relation between logic and thinking. *Psychological Review*, 69, 366-378, 1962.
- Hesse, M. The explanatory function of metaphor. In: *Revolutions and Reconstructions in the Philosophy of Science* by Hesse, M. Brighton, Sussex: Harvester Press Ltd, 1980.
- Hidi, S. and Baird, W. Interestingness - A neglected variable in discourse processing. *Cognitive Science*, 10, 179-194, 1986.
- Hodder, I. ed. *Symbolic and Structural Archaeology*. Cambridge: Cambridge University Press, 1982a.
- Hodder, I. *The Present Past*. London: B.T. Batsford, 1982b.
- Hodder, I. Archaeology in 1984. *Antiquity*, 58, 25-32, 1984.
- Hodder, I. *Reading the past*. Cambridge: Cambridge University Press, 1986.
- Holland, D. and Quinn, N. *Cultural models in language and thought*. Cambridge: Cambridge University Press, 1987.
- Holyoak, K.J. and Thagard, P.R. A computational model of analogical problem solving. To appear in: Ortony, A. and Vosniadou, S. *Similarity and Analogy*.
- Huggett, J. Expert systems in archaeology. In: *Current Issues in Archaeological*

- Computing*, ed. Cooper, M.A. and Richards, J.D. BAR International Series 271, 123-142, 1985.
- Huggett, J. and Baker, K.G. The computerised archaeologist: the development of expert systems. *Science and Archaeology*. 27, 3-7, 1985.
- Hunter, L. Computing: Means and ends in the study of literature. Abstract of paper presented at the conference on Computers and Teaching in the Humanities, Southampton, 1988.
- Huxley, J. and Kettlewell, H.B.D. *Charles Darwin and his world*. London: Thames and Hudson, 1965.
- Janik, A. Tacit knowledge, working life and scientific method. In: Göransson, B. and Josefson, I. eds. *Knowledge, Skill and Artificial Intelligence*. Springer-Verlag: London, 1988.
- Johnson-Laird, P.N. *Mental Models: Towards a Cognitive Science of Language, Inference and Consciousness*. Cambridge: Cambridge University Press, 1983.
- Johnson-Laird, P.N. Reasoning, imagining, and creating. *Bulletin of The British Psychological Society*, 40, 120-129, 1987.
- Josephson, J., Chandrasekaran, B., Smith, J. and Tanner, M. A mechanism for forming composite explanatory hypotheses. *IEEE Trans. on Systems, Man, and Cybernetics*, 17(3), 1987.
- Kahney, H. *Knowledge Engineering*. Open University / Artificial Intelligence Training Pack, PD624KE, Milton Keynes: Open University, 1989.
- Kaplan, S.J. Cooperative responses from a portable natural language query system. *Artificial Intelligence*. 19, 165-187, 1982.
- Keane, M. On drawing analogies when solving problems: a theory and test of solution generation in an analogical problem-solving task. *British Journal of Psychology*, 76, 449-458, 1985.
- Keane, M. Where's the Beef? The absence of Pragmatic Factors in Pragmatic Theories of Analogy. *Proceedings of the 8th European Conference on AI*. London: Pitman Publishing, 1988a.
- Keane, M. Incremental Analogising: Theory & Model. HCRL Tech Rep No 38, The Open University, Milton Keynes 1988b (To appear in Gilhooly, K., Keane, M., Logie, R. and Erdos, G. eds. *Lines of Thought: Reflections on the Psychology of Thinking*. Chichester: Wiley)

- Keane, M. and Brayshaw, M. The Incremental Analogy Machine: A computational model of analogy. HCRL Tech Rep No 33, The Open University, Milton Keynes 1988. (To appear in *Proceedings of the European Working Session on Machine Learning*. London: Pitman)
- Keane, M., Byrne, R.M.J. and Johnson-Laird, P.N. The psychology of arguing. 5th Annual Conference of the Cognitive Psychology Section of the British Psychology Society, September 1988.
- Kedar-Cabelli, S. Purpose-directed analogy. *Proceedings of the Cognitive Science Society Conference*, Irvine, CA, August 1985.
- Kenny, A. *Wittgenstein*. Harmondsworth: Penguin Books, 1973.
- Keravnou, E.T. and Johnson, L. *Competent Expert Systems: A case study in fault diagnosis*. London: Kogan Page Ltd., 1986.
- Kidd, A. What do users ask? Some thoughts on diagnostic advice. In: *Expert Systems '85*, ed. Merry, M. Cambridge: Cambridge University Press, 1985.
- Kim, J. Explanation in Science. In: *The Encyclopedia of Philosophy*. Vol. 3, New York: Macmillan, 1967.
- Kim, J.H. and Pearl, J. CONVINCe: A conversational inference consolidation engine. *IEEE Transactions on Systems, Man, and Cybernetics*. SMC-17, 120-132, 1987.
- Kitcher, P. Explanatory unification. *Philosophy of Science*. 48, 507-531, 1981.
- Knight, J.A. Explanation in a cooperative expert system. *Alvey IKBS Workshop on Explanation*, March 1986.
- Kuipers, B. and Kassirer, J.P. Causal reasoning in medicine: analysis of a protocol. *Cognitive Science*, 8, 363-385, 1984.
- Lacey, A.R. *A Dictionary of Philosophy*. London: Routledge & Kegan Paul, 1976.
- Lagrange, M-S. and Renaud, M. Intelligent knowledge-based systems in archaeology: a computerized simulation of reasoning by means of an expert system. *Computers and the Humanities*, 19, 37-52, 1985.
- Lakoff, G. and Johnson, M. The metaphorical structure of the human conceptual system. *Cognitive Science*, 4, 195-208, 1980a.
- Lakoff, G. and Johnson, M. *Metaphors we live by*. Chicago: University of Chicago Press, 1980b.

- Lakoff, G. *Women, Fire and Dangerous Things*. Chicago: University of Chicago Press, 1987.
- Langlotz, C.P. and Shortliffe, E.H. Adapting a consultation system to critique user plans. *International Journal of Man-Machine Studies*, 19, 479-496, 1983.
- Lebowitz,, M. *Generalization and Memory in an Integrated Understanding System*. Technical Report 186, Dept. of Computer Science, Yale UNiversity, 1980.
- Leith, P. Fundamental errors in legal logic programming. *The Computer Journal*. 29(6), 545-552, 1986.
- Lekson, S.H., Windes, T.C., Stein, J.R. and Judge, W.J. The Chaco Canyon Community *Scientific American*, Vol. 259, No. 1, July 1988.
- Lenat, D.B., Borning, A., McDonald, D., Taylor, C. and Weyer, S. Knoesphere: Building expert systems with encyclopedic knowledge. *Proceedings of the International Joint Conference on AI (IJCAI-83)*, 1983.
- Lenat, D., Prakash, M. and Shepherd, M. CYC: Using common sense knowledge to overcome brittleness and knowledge acquisition bottlenecks. *The AI Magazine*, Winter, 1986.
- Longacre, W.A. *Reconstructing Prehistoric Pueblo Societies*. Albuquerque: University of New Mexico Press, 1970.
- Lowe, D.G. Co-operative structuring of information: the representation of reasoning and debate. *International Journal of Man-Machine Studies*, 23, 97-111, 1985.
- MacCormick, N. *Legal reasoning and Legal Theory*. Oxford: Clarendon Press, 1978.
- Marshall, C.C. Exploring representation problems using hypertext. *Proceedings of the Hypertext 1987 Conference*. November 1987.
- McGuigan, S. and Black, J.B. Creation and comprehension of arguments. In: *Knowledge Structures*, ed. Galambos, J.A., Abelson, R.P. and Black, J.B. Hillsdale, NJ: Lawrence Erlbaum Associates, 1986.
- McGuire, R. Birnbaum, L. and Flowers, M. Opportunistic processing in arguments. *Proceedings of the International Joint Conference on AI (IJCAI-81)*, 1981.
- Meehan, J.R. (1977). TALE-SPIN, an interactive program that writes stories. *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI-77)*.

- Megaw, J.V.S and Simpson, D.D.A. *Introduction to British Prehistory*, Leicester: Leicester University Press, 1979.
- Miall, D.S. Design principles for the computer assisted management of literary response. *Future Computing Systems*. 1(2), 217-231, 1986.
- Miall, D.S. An expert systems approach to the interpretation of literary structure. Paper prepared for the 2nd Anglo-French Round Table "Exploring the Foundations of Interpretation in the Humanities: The Role of Artificial Intelligence." Bourne, Lincs. Sept 14-17 1988a.
- Miall, D.S. Rethinking learning methods. Abstract of paper presented at the conference on Computers and Teaching in the Humanities, Southampton, 1988b.
- Midgley, M. *Beast and Man: The Roots of Human Nature*. London: Methuen and Co., 1980.
- Miller, P.L. ATTENDING: critiquing a physician's management plan. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAM-5 (5), 449-461, 1983.
- Minsky, M. A framework for representing knowledge. In. ed. Winston, P.H. *The Psychology of Computer Vision..* New York: McGraw-Hill, 1975.
- Mitchell, T.M., Keller, R.M. and Kedar-Cabelli, S.T. Explanation-based generalization: a unifying view. *Machine Learning*, 1(1), 47-80, 1986.
- Moor, J.H. Three myths of computer science. *British Journal for Philosophy of Science*. 29, 213-222, 1978.
- Mott, P. and Brooke, S. A graphical inference system. *Expert Systems*, 4(2), 106-117, 1987.
- Natali-Smit, J. Seshat et l'analyse poétique: a propos des <<Chats>> de Baudelaire. In: Gardin et al., 1981.
- Neches, R., Swartout, W.R. and Moore, J.D. Enhanced maintenance and explanation of expert systems through explicit models of their development. *IEEE Trans. on Software Engineering*, SE-11, 1337-1351, 1985.
- Newbold, R.F. Polysemy and authority in the late Roman Empire. *Semiotica*, 71(3/4), 227-242, 1988.
- Newton, K.M. *In Defence of Literary Interpretation: Theory and Practice*. London: Macmillan, 1986.

- O'Keefe, D.J. Two concepts of argument. *The Journal of the American Forensic Association*. Vol 13 (3), 121-128, 1977.
- Olsen, S. H. *The structure of literary understanding*. Cambridge: Cambridge University Press, 1978.
- Patel, J. and Stutt, A. Kiva: an archaeological interpreter. Technical Report No. 35, HCRL, The Open University, 1988.
- Patel, J. and Stutt, A. Reducing uncertainty in archaeological interpretation using expectation based reasoning. In: eds. Kelly, B. and Rector, A.L. *Research and Development in Expert Systems V*. Cambridge: Cambridge University Press. (The Proceedings of the *Expert Systems 88* conference, Brighton), 1989.
- Perkins, D.N., Allen, R. and Hafner, J. Difficulties in everyday reasoning. Paper prepared for Conference On Thinking, The University of the South Pacific, Fiji, 1982.
- Pfau, M., Thomas, D.A. and Ulrich, W. *Debate and Argument: A systems approach to advocacy*. Glenview, Illinois: Scott, Foresman and Co., 1987.
- Piggott, S. 'The Early Bronze Age in Wessex', *Proc. Prehistoric Society*, 4, 1938.
- Pirie, D.B. *How to write critical essays: a guide for students of literature*. London: Methuen, 1985.
- Plato, *Symposium and other Dialogues*. Translated: Warrington, J. et al. London: Dent, 1964.
- Platt, J.R. Strong inference. *Science*, vol. 146, no. 3642, 1964.
- Popper, K. *Conjectures and Refutations*. London: Routledge and Kegan Paul, 1963.
- Potter, R.G. Literary criticism and literary computing: The difficulties of a synthesis. *Computers and the Humanities*, 22 91-7 1988.
- Rector, A.L., Newton, P.D. and Marsden, P.H. What kind of system does an expert need? In: *People and computers*. ed. Johnson, P and Cook, S. Cambridge: Cambridge University Press, 1985.
- Reichman, R. Analogies in spontaneous discourse. *Proceedings of the Annual Meeting of the Association Computational Linguistics*, 63-69, 1981.
- Reichman, R. *Getting computers to talk like you and me*. Cambridge, Mass.: MIT Press, 1985.
- Reichman-Adar, R. Extended person-machine interface. *Artificial Intelligence*, 22,

157-218, 1984.

Reilly, P. Computers in field archaeology: agents of change? In: *Current Issues in Archaeological Computing* ed. Cooper, M.A. and Richards, BAR International Series 271, 63-78, 1985.

Richer, M.H and Clancey, W. J. GUIDON-WATCH: A graphical interface for viewing a knowledge-based system. *IEEE Computer Graphics and Applications*. 11(5), 51-64, 1985.

Rips, L.J. Cognitive processes in propositional reasoning. *Psychological Review*, 90(1), 38-71, 1983.

Rowland, R.C. The influence of purpose on fields of argument. *The Journal of the American Forensic Association*. vol 18, 228-245, 1982.

Rowland, R.C. On defining argument. *Philosophy and Rhetoric*. 20(3), 140-159, 1987.

Sabloff, J.A. *Simulations in Archaeology*, Albuquerque: University of New Mexico Press, 1981.

Sadock, J.M. Modus Brevis: the truncated argument. Papers from the Chicago Linguistics Society, 13th Regional Meeting, 545-554, 1977.

Saunders, I. Defending interpretation. (A review of Newton, 1986). *Textual Practice*, 1(3), 356-360, 1987.

Schank, R. Interestingness: Controlling inferences. *Artificial Intelligence*, 12, 273-297, 1979.

Schank, R. Explanation: a first pass. In: Kolodner, J.L. and Riesbeck, C.K. *Experience, Memory and Reasoning*. New Jersey: Lawrence Erlbaum Assoc., 1986a.

Schank, R.C. *Explanation Patterns: Understanding Mechanically and Creatively*. Hillsdale, N.J.: Lawrence Erlbaum Associates, 1986b.

Schiffer, M.B. *Behavioral Archaeology*. New York: Academic Press, 1976.

Schwartz, H.J. Computer software for critical thinking. *Computers and Education*. Vol 12(1), 163-168, 1988.

Scriven, M. *Reasoning*. New York: McGraw-Hill, 1976.

Searle, J.R. Minds, Brains, and Programs. *The Behavioral and Brain Sciences*, Vol 3, 1980.

- Searle, J.R. Minds and brains without programs. In: Blakemore, C. and Greenfield, S. *Mindwaves*. Oxford: B. Blackwell, 1987.
- Sergot, M.J., Sadri, F., Kowalski, R.A., Kriwaczek, F., Hammond, P. and Cory, H.T. The British Nationality Act as a logic program. *Communications of the ACM*, 29(5), 370-386, 1986.
- Shackel, B. Plans and initial progress with BLEND - an electronic network communication experiment. *International Journal of Man-Machine Studies*, 17, 225-23, 1982.
- Shanks, M and Tilley, C. Ideology, symbolic power and ritual communication: a reinterpretation of Neolithic mortuary practices. In: Hodder 1982.
- Shanks, M and Tilley, C. *Re-Constructing Archaeology*. Cambridge: Cambridge University Press, 1987.
- Shennan, S. Ideology, change and the European Early Bronze Age. In: Hodder 1982.
- Shennan, S. Why should we believe archaeological interpretations? Paper prepared for the 2nd Anglo-French Round Table "Exploring the Foundations of Interpretation in the Humanities: The Role of Artificial Intelligence", Lincolnshire, September 1988.
- Shortliffe, E.H. *Computer-Based Medical Consultations: MYCIN*. New York: Elsevier, 1976.
- Slatin, J. Text and hypertext: computers and the teaching of poetry. Abstract of paper presented at the conference on Computers and Teaching in the Humanities, Southampton, 1988.
- Smith, R.G., Mitchell, T.M., Winston, H.A. and Buchanan, B.G. Representation and use of explicit justifications for knowledge base refinement. *Proceedings of the Ninth International Joint Conference on Artificial Intelligence (IJCAI-85)*, 1985.
- Sperber, D. and Wilson, D. Précis of *Relevance: Communication and Cognition*. Behavioral and Brain Sciences, 10, 697-710, 1987.
- Stefik, M. The Next Knowledge Medium. *AI Magazine*, 7(1), 34-46, 1986.
- Stefik, M., Aikens, J., Balzer, R., Bendit, J., Birnbaum, L. Hayes-Roth, F. and Sacerdoti, E. The organisation of expert systems, a tutorial. *Artificial Intelligence*, 18(2), 135-173, 1982.
- Stefik, M., Foster, G., Bobrow, D.G., Kahn, K., Lanning, S. and Suchman, L. Beyond the chalkboard: computer support for collaboration and problem solving in meetings. *Communications of the ACM*, 30(1), 32-47, 1987.

- Stiles, D. Ethnoarchaeology: A discussion of methods and applications. *Man*, 12, 87-103, 1977.
- Stutt, A. *Second generation expert systems, explanations, arguments and archaeology*. Technical Report No. 25, HCRL, The Open University, 1987.
- Stutt, A. Second generation expert systems, explanations, arguments and archaeology. Paper presented at the Computer Applications in Archaeology Conference at Birmingham, March, Published in: ed. Rahtz, S.P.Q. *Computer and Quantitative Methods in Archaeology*. Oxford: BAR International Series 446, 1988a.
- Stutt, A. Argument support programs: machines for generating interpretations. To be published in the Proceedings of the Conference on Computers and Teaching in the Humanities, Southampton, 1988b.
- Stutt, A. The use of artificial intelligence techniques to support archaeological reasoning. To be published in the Proceedings of L'Atelier Européen, Valbonne, 1989
- Swartout, W.R. XPLAIN: a system for creating and explaining expert consulting programs. *Artificial Intelligence*, 21, 285-325, 1983.
- Taylor, J. Why novices will find learning Prolog hard. *Proceedings of the Sixth European Conference on Artificial Intelligence (ECAI-84)*, 1984.
- Thagard, P. The best explanation: criteria for theory choice. *Journal of Philosophy*, 75, 76-92, 1978.
- Thagard, P. Programs, Theories, and Models. *Proc. Conf. of the Cognitive Science Society*, 155-157, 1982.
- Thagard, P. Explanatory Coherence. Technical report CSL Report 16, Cognitive Science Laboratory, Princeton University, March 1988a.
- Thagard, P. *Computational Philosophy of Science*. Cambridge, Mass.: MIT Press, 1988b.
- Thagard, P. Explanatory Coherence. To appear in: *Behavioral and Brain Sciences*.
- Thouless, R.H. *Straight and crooked thinking*. London: Pan Books Ltd 1974 (first published Hodder and Stoughton 1930).
- Toulmin, S. *The Uses of Argument*. Cambridge: Cambridge University Press, 1958.
- Toulmin, S., Rieke, R. and Janik, A. *An Introduction to Reasoning*. New York: Macmillan Publishing Co, 1979.

- Trigg, R. H. and Weiser, M. TEXTNET: A network-based approach to text handling. *ACM Transactions on Office Information Systems*. 4(1), 1-23, 1986.
- Twain, M. *The Adventures of Huckleberry Finn*. London: Penguin Books Ltd, 1987.
- VanLehn, K., Brown, J.S. and Greeno, J. Competitive argumentation in computational theories of cognition. In: *Method and Tactics in Cognitive Science*. eds: Kintsch, W., Miller, J.R. and Polson, P.G. Hillsdale, N.J.: Lawrence Erlbaum Associates, 1984.
- VanLehn, K. *Theory Reform Caused by an Argumentation Tool*. Technical Report ISL-11, Xerox Corporation, Palo Alto, California, July 1985.
- Vitali, V. and Lagrange, M-S. An expert system for the provenance determination of archaeological ceramics based on INAA data. Paper presented at the Computer Applications in Archaeology Conference at Birmingham, March, Published in: ed. Rahtz, S.P.Q. *Computer and Quantitative Methods in Archaeology*. Oxford: BAR International Series 446, 1988.
- Waddell, L.A. *Tibetan Buddhism*. New York: Dover Publications Inc., 1972.
- Wainwright, G.J. Religion and settlement in Wessex, 3000-1700 bc. In: ed. Fowler, P.J. *Recent Work in Rural Archaeology*. Bradford-on-Avon: Moonraker Press, 1975.
- Wallis, J. and Shortliffe, E. H. Customized methods using causal knowledge. *Methods of Information in Medicine*, 21, 127-136, 1982.
- Ward, K. *John Keats: The making of a Poet*. London: Mercury Books, 1966.
- Watson, J.D. *The Double Helix*. London: Weidenfeld and Nicolson, 1968.
- Weiner, J.L. BLAH: a system which explains its reasoning. *Artificial Intelligence*, 15, 19-48, 1980.
- Weisberg, R. *Creativity: Genius and other myths*. New York: W.H. Freeman and Co., 1986.
- Weizenbaum, J. *Computer Power and Human Reason*. Freeman and Co. 1976. (Harmondsworth: Penguin Books Ltd 1984)
- Wielinga, B.J. and Breuker, J.A. Models of expertise. *Proceedings of the Seventh European Conference on Artificial Intelligence (ECAI-86)*. 1986.
- Wilks, Y. and Bien, J. Beliefs, points of view, and multiple environments. *Cognitive Science*, 7, 95-119, 1983.
- Willard, C.A. Propositional argument is to argument what talking about passion is to

- passion. *The Journal of the American Forensic Association*. vol 16. 21-28. 1979a.
- Willard, C.A. The epistemic functions of argument: reasoning and decision-making from a constructivist/interactionist point of view. (Parts I and II) *The Journal of the American Forensic Association*. vol 15, 1979b.
- Winch, P. Understanding a primitive society. In: Wilson, B.R. *Rationality*. Oxford: Basil Blackwell, 1970.
- Winograd, T. What does it mean to understand language? *Cognitive Science*, 4, 209-241, 1980.
- Winograd, T. and Flores, F. (1986). *Understanding Computers and Cognition*, NJ: Ablex Publishing Corp.
- Winston, P.H. Learning and reasoning by analogy. *Communications of the ACM*, 23(12), 1980.
- Wittgenstein, L. *Philosophical Investigations*. Oxford: Blackwell, 1953.
- Woodward, F.L. Trans. *Some Sayings of the Buddha*. Oxford: Oxford University Press, 1973.
- Wordern, R.P., Foote, M.H., Knight, J.A. and Anderson, S.K. Cooperative expert systems. *Proceedings of the Seventh European Conference on Artificial Intelligence (ECAI-86)*. 1986.
- Wylie, A. 'Simple' analogy and the role of relevance assumptions: Implications of archaeological practice. Unpublished paper. University of Western Ontario, 1987.
- Zadeh, L. A. Fuzzy logic. *Computer*, 21(4), 83-92, 1988.

-o0o-

Appendix I - Selections from the Prolog code for ASParch

```
/* main control */
```

```
maincontrol:-
```

```
  init_run1, init_run2,
  retract_sys_kb,
  introduction,
  message([' Do you want to go on? ']),
  get_field(F), retractall(current_field(_)), assert(current_field(F)),
  get_topic(T), retractall(current_topic(_)), assert(current_topic(T)),
  first_move.
```

```
%% NB in what follows the current roots will change during interaction
```

```
%% USUS form
```

```
first_move:-
```

```
  yesno(['Do you want to go first?']),
  ask_for_argument,          %% get user's interpretation
  current_user_root(R),
  udisplay_tree('user interpretation1',R),
  look_for_argument,        %% get system interpretation
  retractall(previous_user_root(_)),assert(previous_user_root(R)),
  current_sys_root(R1),
  retractall(previous_sys_root(_)),assert(previous_sys_root(R1)),
  udisplay_tree('system interpretation1',R1),
  ask_for_user_response,    %% get user's response
  current_user_root(R2),
  udisplay_tree('user interpretation2',R2),
  attack_or_defend_1,       %% get system response
  current_sys_root(R3),
  udisplay_tree('system interpretation2',R3),
  !.
```

```
%% SUSU
```

```
first_move:-
```

```
  look_for_initial_argument, %% get system interpretation
  current_sys_root(R1),
  udisplay_tree('system interpretation1',R1),
  message([' Your turn ']),
  retractall(previous_sys_root(_)),assert(previous_sys_root(R1)),
  ask_for_argument, %% get user's interpretation
  current_user_root(R2),
  udisplay_tree('user interpretation1',R2),
  attack_or_defend_2, %% get system response
  retractall(previous_user_root(_)),assert(previous_user_root(R2)),
  current_sys_root(R3),
  udisplay_tree('system interpretation2',R3),
  ask_for_user_response, %% get user's response - use parameter to take in prop from menu!!
  current_user_root(R4),
  udisplay_tree('user interpretation2',R4),!.
```

```
introduction:-
```

```
  create_dialogue_window('ASPA dialogue'),
```

```

nl('ASPA dialogue'),nl('ASPA dialogue'),
write('ASPA dialogue','Welcome to the Argument Support Program for Archaeology'),
nl('ASPA dialogue'),nl('ASPA dialogue'),
write('ASPA dialogue','The interaction takes the form of a formal debate in which'),
nl('ASPA dialogue'),nl('ASPA dialogue'),
write('ASPA dialogue','both system an user have two goes'),
nl('ASPA dialogue'),nl('ASPA dialogue'),
write('ASPA dialogue','1. To give an archaeological interpretation'),
nl('ASPA dialogue'),nl('ASPA dialogue'),
write('ASPA dialogue','2. to defend and/or criticise the others argument').

%% deletes windows for new run
init_run1:-
    wkill('system interpretation1'),
    wkill('system interpretation2'),
    wkill('user interpretation1'),
    wkill('user interpretation2'),
    wkill('ASPA dialogue').

init_run2:-
    retractall(previous_user_root(_)),
    assert(previous_user_root(none)),
    retractall(previous_sys_root(_)),
    assert(previous_sys_root(none)).

get_previous_sys_root(R):-
    previous_sys_root(none),
    current_sys_root(R).
get_previous_sys_root(R):-
    previous_sys_root(R).

get_previous_user_root(R):-
    previous_user_root(none),
    current_user_root(R).
get_previous_user_root(R):-
    previous_user_root(R).

%% get system argument
look_for_argument:-
    fc,!,
    current_user_root(R),
    my_fetch(R, claim,the Att of Obj is Val),
    (prove(the Att of Obj is Val); %% i.e. enforces relevance to user claim
     prove(the Att of Obj is not(Val))),
    root_sys_arg(the Att of Obj is Val),!. %%convert production trace to arg nodes
look_for_argument:-
    message(['Aspa is unable to find a relevant argument']),fail.

look_for_initial_argument:-
    fchain, % this calls fc and even if it fails it returns true

```

```

current_topic(the Att of Obj),
prove(the Att of Obj is Val1), %% i.e. enforces relevance to topic
root_sys_arg(the Att of Obj is Val1),!. %%convert production trace to arg nodes
look_for_initial_argument:-
    message(['Aspa is unable to find a relevant argument']),fail.

```

```

% user responds attack
ask_for_user_response:-
    menu1(S), S=['attack sys arg'],
    menu2(Attack_type),
    wfront('system interpretation1'),
    message(['Please focus on a single proposition in the system arg']),
    get_previous_sys_root(R),
    current_user_root(R0), % old root
    ask_for_argument,
    current_user_root(R1),
    note(the response_type of R1 is Attack_type),
    note(the external_attacks of R1 is R),
    note(the external_supports of R1 is R0),
    note(the external_attacked_by of R is R1),
    note(the external_supported_by of R0 is R1),!.

```

```

% user responds defend
ask_for_user_response:-
    menu3(Defence_type),
    wfront('user interpretation1'),
    message(['Please focus on a single proposition in the user arg']),
    current_user_root(R), % old root
    get_previous_sys_root(R0),
    ask_for_argument,
    current_user_root(R1), % new root
    note(the response_type of R1 is Defence_type),
    note(the external_supports of R1 is R),
    note(the external_attacks of R1 is R0),
    note(the external_supported_by of R is R1),
    note(the external_attacked_by of R0 is R1),!.

```

```

% failed to get response
ask_for_user_response:-
    message(['Aspa is quitting because there has been no user response']),fail.

```

```

menu1(Selected):-
scroll_menu([' Please select your response type '],
    [ 'attack sys arg',
      'defend own arg',
      'alter arg'], [],Selected).
menu2(Selected):-
scroll_menu([' Please select your attack type '],
    [ 'attack sys claim',
      'attack sys ground'],[],Selected).
menu3(Selected):-
scroll_menu([' Please select your defence type '],
    [ 'defend user claim',

```

```

'defend user ground'],[],Selected).

%% USUS
%% defend i.e. no weak spots in either current or previous user arg
attack_or_defend_1:-
    current_user_root(R),
    find_weak_spots(R,Spots),
    Spots = [],
    previous_user_root(PR),
    find_weak_spots(PR,PSpots),
    PSpots = [],
    support_sys_arg,!

%% attack weak spots since previous user arg had some
attack_or_defend_1:-
    previous_user_root(R), % or sys arg!!! i.e. as part of reasoning
    find_weak_spots(R,Spots),
    Spots = [],
    current_user_root(PR),
    find_weak_spots(PR,PSpots),
    attack_weak_spots(PR,PSpots),!.

%% attack previous user argument
attack_or_defend_1:-
    previous_user_root(R),
    find_weak_spots(R,Spots), !,
    attack_weak_spots(R,Spots).

%% SUSU
%% defend i.e. no weak spots in either current or previous user arg
attack_or_defend_2:-
    current_user_root(R),
    find_weak_spots(R,Spots),
    Spots = [],
    support_sys_arg,!

%% attack previous user argument
attack_or_defend_2:-
    current_user_root(R),
    find_weak_spots(R,Spots), !,
    attack_weak_spots(R,Spots).

find_weak_spots(none,[]):-!.
find_weak_spots(R,Spots):-
    examine_node(R,Spots1),
    filter(Spots1,Spots),!.

examine_grounds([],[]).
examine_grounds([[]],[]).
examine_grounds({S1 | Rest},[Spot|RestSpots]):-
    examine_node(S1, Spot),
    examine_grounds(Rest,RestSpots). % rest of grounds here

```

```

examine_node(Node, [[Node , Status] | Rest]):-
    my_fetch(Node,claim,Proposition),
    my_fetch(Node,warrant,Warrant),
    my_fetch(Node,supported_by,Supports),
    my_fetch(Node,grounds,Grounds),
    weakstatus(Proposition,Grounds,Warrant,Status),
    examine_grounds(Supports,Rest). %% sub-grounds of this sub-ground

filter(S,S1):-
    squash(S,FlatS),
    filter1(FlatS,S1).

filter1([],[]).
filter1([Node,ok|Rest], Rest1):-
    filter1(Rest,Rest1).
filter1([Node,Status|Rest], [Node,Status|Rest1]):-
    filter1(Rest,Rest1).

attack_weak_spots(_,[]).
attack_weak_spots(R,[Node, Status|Rest]):- %% R= node attacked
    attack_weak_spot_main(R,Node,Status),
    attack_weak_spots(R,Rest).

/* If analogy and the field is archaeology then deal with in a particular way */
attack_weak_spot_main(R,Node,analogy_ground):-
    current_field(archaeology),
    find_analogy_type(Node, AType,Atts, Selected),!,
    deal_with_analogy(R,Node,AType,Atts,Selected),!.

attack_weak_spot_main(R,Node,analogy_ground):-
    deal_with_analogy(R,Node,simple_analogy,Atts, Selected).

attack_weak_spot_main(R,Node,arguable_principle(P)):-
    deal_with_principle(R,Node,arguable_principle(P)).

attack_weak_spot_main(R,Node,selfcontradictory(P)):-
    deal_with_selfcontradictory(R,Node,selfcontradictory(P)).

attack_weak_spot_main(R,Node,_):-
    attack_weak_spot(R,Node,_,_,_).

deal_with_analogy(R,Node, simple_analogy,Atts, Selected):-
    attack_weak_spot(R,Node,simple_analogy,Atts, Selected).

deal_with_analogy(R,Node, historical_analogy,Atts, Selected):-
    attack_historical_link(R,Node).

deal_with_analogy(R,Node, relational_analogy,Atts, Selected):-
    attack_relativity(R,Node).

deal_with_principle(R,Node,arguable_principle(P)):-
    attack_weak_spot(R,Node,arguable_principle(P)).

deal_with_selfcontradictory(R,Node,selfcontradictory(P)):-

```

```

    attack_weak_spot(R,Node,selfcontradictory(P)).

find_analogy_type(Node, AType,Attlist,Selected):-
    my_fetch(Node, claim, the analogue of X is Y),
    get_artifact_atts_s(X,Attlist),
    scroll_menu([' Please select the matching attributes of, X, ' and ', Y],
                Attlist,[],Selected),!,
    get_atype(X,Y,Attlist,Selected,AType).

get_atype(X,Y,Atts,Selectedatts,historical_analogy):-
    prompt_read(['Is there a historical link between', X, ' and ', Y],Yes), Yes = yes,!.

get_atype(X,Y,Atts,Selectedatts,relational_analogy):-
    prompt_read(['Is there a relational link between the attributes of, X, ' and ', Y],Yes),
    Yes = yes,!.

get_atype(X,Y,Atts,Selectedatts,simple_analogy):-
    Atts = Selectedatts,!.

get_atype(X,Y,Atts,Selectedatts,simple_analogy):-
    subset(Selectedatts,Atts),!.

%% 1. look for saliency
attack_weak_spot(Nodeattacked,Node,simple_analogy,Atts, Selected):-
    nl,write(' ASPA trying to spot failure in saliency...'),
    my_fetch(Nodeattacked,claim,C),
    my_fetch(Node,claim,the analogue of A1 is A2),
    current_field(F),
    unmatched_salient_attribute(F,Atts,Selected,Salient),
    make_sys_arg(saliency,C,[not the analogue of A1 is A2],Salient),
    nl,write(' ASPA succeeded in finding failure in saliency!'),nl;
    nl,write(' ASPA has failed to detect error in saliency!'), fail.

%% 2. look for another analogy which is contrary
attack_weak_spot(Nodeattacked,Node,simple_analogy,Atts, Selected):-
    nl,write(' ASPA trying to find alternative analogy...'),
    my_fetch(Nodeattacked,claim,C),
    my_fetch(Node,claim,the analogue of A1 is A2),
    find_another_analogue(A1,A2,Anotheranalogue),
    current_field(F),
    contrary_analogue(F,Anotheranalogue,A2),
    make_sys_arg(alt_analogue,C,[the contrary_analogue of A1 is Anotheranalogue]),
    nl,write(' ASPA succeeded in finding alternative analogy!'),nl;
    nl,write(' ASPA has failed to find alternative analogy!'), fail.

%% 3. deal with weak principle
attack_weak_spot(Nodeattacked,Node,arguable_principle(P)):-
    nl,write(' ASPA attacking principle...'), write(P),
    my_fetch(Nodeattacked,claim,C),
    my_fetch(Node,claim,C2),
    make_sys_arg(principle,C,[not C2],P),
    nl,write(' ASPA successful in attack on principle!'),nl;
    nl,write(' ASPA has failed in attack on principle!'), fail.

```

```

%% 4. deal with selfcontradictory
attack_weak_spot(Nodeattacked,Node,selfcontradictory(P)):-
    nl,write(' ASPA attacking self contradiction...'), write(P),
    my_fetch(Nodeattacked,claim,C),
    my_fetch(Node,claim,C2),
    make_sys_arg(selfcontradiction,C,[not C2],P),
    nl,write(' ASPA successful in attack on self contradiction!'),nl;
    nl,write(' ASPA has failed in attack on self contradiction!'), fail.

attack_weak_spot(Node,_,_,_):-
    message(['attacking user claim']).

make_sys_arg(saliency,C,G,Salient):-
    initial_make_arg(Arg,Arg2,Arg3,Sysroot,Userroot),
    note((Arg instance_of argument with
        claim: not(C),
        grounds: G,
        warrant: 'saliency is good',
        supported_by: [Arg2,[]],
        external_supports: Sysroot,
        external_attacks: Userroot)),
    G = [Gr], Gr= not(the analogue of A1 is A2),
    note((Arg2 instance_of argument with
        claim: (Gr),
        grounds:[the sal_feature(Salient) of A2 is unmatched],
        warrant: unknown,
        supported_by: [Arg3,[]])),
    note((Arg3 instance_of argument with
        claim: (the sal_feature(Salient) of A2 is unmatched),
        grounds:[],
        warrant: unknown,
        supported_by: [])),
    note(the external_supported_by of Sysroot is Arg),
    note(the external_attacked_by of Userroot is Arg),!.

make_sys_arg(alt_analogue,C,G):-
    initial_make_arg(Arg,Arg2,Arg3,Sysroot,Userroot),
    note((Arg instance_of argument with
        claim: not(C),
        grounds: G,
        warrant: 'contrary analogues',
        supported_by: [Arg2,[]],
        external_supports: Sysroot,
        external_attacks: Userroot)),
    G = [Gr],
    note((Arg2 instance_of argument with
        claim: (Gr),
        grounds:[],
        warrant: unknown,
        supported_by: [])),
    note(the external_supported_by of Sysroot is Arg),
    note(the external_attacked_by of Userroot is Arg),!.

```

```

make_sys_arg(principle,C,G,Pr):-
  initial_make_arg(Arg,Arg2,Arg3,Sysroot,Userroot),
  note((Arg instance_of argument with
    claim: not(C),
    grounds: G,
    warrant: 'weak principle',
    supported_by: [Arg2,[]],
    external_supports: Sysroot,
    external_attacks: Userroot)),
  G = [Gr], Gr = (not G1),
  note((Arg2 instance_of argument with
    claim: (Gr),
    grounds:[the support_principle(Pr) of G1 is weak],
    warrant: unknown,
    supported_by: [Arg3,[]])),
  note((Arg3 instance_of argument with
    claim: (the support_principle(Pr) of G1 is weak),
    grounds:[],
    warrant: unknown,
    supported_by: [])),
  note(the external_supported_by of Sysroot is Arg),
  note(the external_attacked_by of Userroot is Arg),!.

```

```

make_sys_arg(selfcontradiction,C,G,P):-
  initial_make_arg(Arg,Arg2,Arg3,Sysroot,Userroot),
  note((Arg instance_of argument with
    claim: not(C),
    grounds: G,
    warrant: 'selfcontradiction',
    supported_by: [Arg2,[]],
    external_supports: Sysroot,
    external_attacks: Userroot)),
  G = [Gr], Gr = (not G1),
  note((Arg2 instance_of argument with
    claim: (Gr),
    grounds:[the claim of G1 is selfcontradictory_with(P)],
    warrant: unknown,
    supported_by: [Arg3,[]])),
  note((Arg3 instance_of argument with
    claim: (the claim of G1 is selfcontradictory_with(P)),
    grounds:[],
    warrant: unknown,
    supported_by: [])),
  note(the external_supported_by of Sysroot is Arg),
  note(the external_attacked_by of Userroot is Arg),!.

```

```

unmatched_salient_attribute(F,Atts,Selected,SF):-
  on(SF,Atts),
  not on(SF,Selected),
  salient_features_for(F, SF).

```

```

initial_make_arg(Arg,Arg2,Arg3,Sysroot,Userroot):-
  gensym(sysarg,Arg),

```



```
gensym(sysarg,Arg2),
gensym(sysarg,Arg3),
current_sys_root(Sysroot),
get_previous_user_root(Userroot),
retractall(current_sys_root(_)),
assert(current_sys_root(Arg)).
```

```
salient_features_for(archaeology,date).
salient_features_for(archaeology,angle).
salient_features_for(F,Sf):-
    theory_for(F,T),
    salient_by(T,Sf).
```

```
theory_for(archaeology, marxism).
salient_by(marxism, power).
```

```
find_another_analogue(A1,A2,Anotheranalogue):-
    (A1 instance_of Class with Body1),
    assert_sys_kb,
    sys_belief((Anotheranalogue instance_of Class with Body2)),
    not A1 = Anotheranalogue,
    has_attributes_s(A1, List1),
    has_attributes_s(Anotheranalogue, List2),
    same_atts_s(List1, List2).
```

```
%% not the same uses for now!!
contrary_analogue(F,Anotheranalogue,A2):-
    my_fetch(Anotheranalogue,use,U),
    my_fetch(A2,use,U2),
    not U = U2.
```

```
support_sys_arg:-
    message(['supporting sys arg ....']).
```

```

/* weak status rules */

%% rules for weak links in interpretation %%

%% analogy
weakstatus(P,G,W,analogy_ground):-
    P = (the analogue of X is Y).

%% self-contradictory
weakstatus(P,G,W,selfcontradictory(X)):-
    assert_user_kb,
    user_belief(X),
    contrary(X,P).

weakstatus(P,G,W,selfcontradictory(X)):-
    user_belief(X),
    can_forward_derive(X1,X),
    contrary(X1,P).

weakstatus(P,G,W,selfcontradictory(X)):-
    user_belief(X),
    can_forward_derive(X2,P),
    contrary(X,X2).

weakstatus(P,G,W,selfcontradictory(X)):-
    user_belief(X),
    can_forward_derive(X1,X),
    can_forward_derive(X2,P),
    contrary(X1,X2).

%% arguable principle = level shift
weakstatus(P,G,DummyW,arguable_principle(Pr)):-
    not G = [],
    G = [G1],
    W = (G1 -> P),
    backing_for(W,B),
    min_valued_principle(B,Pr).

%% ok status
weakstatus(P,G,W,ok).

```

```

/* dialogue control */

ask_for_argument:-
    get_cg(C,List_of_grounds),
    gensym(userarg,Arg),
    retractall(current_user_root(_)),assert(current_user_root(Arg)),
    note((Arg instance_of argument with
        claim: C, grounds:List_of_grounds,warrant:unknown,supported_by: [])),
    recurse_on_grounds(Arg,List_of_grounds),!.

recurse_on_grounds(_,[ ]).
recurse_on_grounds(Arg,[H|T]):-
    message(['Taking', H , 'as a sub_claim, give grounds']),
    get_grounds(List_of_grounds),
    gensym(userarg,NewArg),
    note((NewArg instance_of argument with
        claim: H, grounds:List_of_grounds,warrant:unknown,
        supports: Arg, supported_by: [])),
    prove(all supported_by of Arg are OldSupports),
    note((the supported_by of Arg is [NewArg | OldSupports])),
    recurse_on_grounds(NewArg,List_of_grounds),
    recurse_on_grounds(Arg,T),!.

get_cg(C,G):-
    get_claim(C),
    get_grounds(G).

get_field(F):-
    prompt_read([ 'Field? '],F).
get_topic(T):-
    prompt_read([ 'Topic? '],T).
get_claim(C):-
    prompt_read([ 'Claim? '],C).
get_grounds([G|T]):-
    get_ground(G), not G = stop,get_grounds(T),!.
get_grounds([ ]).

get_ground(G):-
    prompt_read([ 'Ground (type <stop> to end) '],G).

```

/* assessment code */

overall_assess_arg(Value):-

```

    get_user_root(R1), %% i.e. user's first argument
    get_user_aux(R2), %% i.e. user's second argument
    assess_argument(R1, Value1),
    nl,write(' user argument 1 > '),write(Value1),nl,
    assess_argument(R2, Value2),
    nl,write(' user argument 2 > '),write(Value2),nl,
    get_sys_root(R3),
    get_sys_aux(R4), %% i.e. sys second argument
    assess_argument(R3, Value3),
    nl,write(' system argument 1 > '),write(Value3),nl,
    assess_argument(R4, Value4),
    nl,write(' system argument 2 > '),write(Value4),nl,
    combine_overall_values(Value1, Value2, OValue1),
    nl, write('overall value for user argument > '), write(OValue1),
    combine_overall_values(Value3, Value4, OValue2),
    nl, write('overall value for system argument > '),write(OValue2),nl,nl,
    winner(OValue1, OValue2, Value).

```

assess_argument(Arg, Value):-

```

    assess_root_arg(Arg, Value1),
    nl,write('    root value of '), write(Arg), write('is > '), write(Value1),
    (get_directly_attacked_by(Arg, External_attacked_by),
    assess_attacked_by(Arg, External_attacked_by, Value2);
    Value2 = 0),
    nl,write('    attacked by value of '), write(Arg), write('is > '),
    write(Value2),
    %% recursively assess this root
    (get_directly_supported_by(Arg, External_supported_by),
    assess_supported_by(Arg, External_supported_by, Value3);
    Value3 = 0),
    nl,write('    supported by value of '), write(Arg), write('is > '),
    write(Value3),
    %% recursively assess this root
    combine_argvalues(Value1, Value2, Value3, Value).

```

assess_root_arg(Arg, Value):-

```

    get_total_nodes(Arg, Nodecount),
    nl, write('        the total nodes of '), write(Arg), write('is > '),
    write(Nodecount),
    get_weak_nodes(Arg, Weaknodecount),
    nl, write('        the weak nodes of '), write(Arg), write('is > '),
    write(Weaknodecount), !,
    check_ratio(Nodecount, Weaknodecount, Value).

```

%% If both same then argument full of weak nodes

check_ratio(N,W,-V):-

```

    N = W,
    V is 0.

```

%% If weakcount is greater than half possible nodes then poor

check_ratio(N,W, V):-

 N1 is (N/2),

 W > N1,

 V is 3.

%% If weakcount is equal to half possible nodes then middling

check_ratio(N,W, V):-

 N1 is (N/2),

 N1 = W,

 V is 5.

%% If weakcount is 0 then strong

check_ratio(N,W, V):-

 W = 0,

 V is 10.

%% If weakcount is less than half possible nodes then reasonable

check_ratio(N,W, V):-

 N1 is (N/2),

 N1 > W,

 V is 7.

%% Formula F1 above

combine_overall_values(V1, V2, OV):-

 OV is V1 + V2.

%% Formula F2 above

combine_argvalues(V1, V2, V3, V):-

 V is (V1 + V3) - V2.

winner(OValue1, OValue2, Value):-

 max(OValue1, OValue2, OValue1),

 Value = user.

winner(OValue1, OValue2, system).

max(A,B,A):-

 A > B.

max(A,B,B).

get_user_root(R):-

 previous_user_root(R).

get_user_aux(R):-

 current_user_root(R).

get_sys_root(R):-

 previous_sys_root(R).

get_sys_aux(R):-

 current_sys_root(R).

```
get_directly_attacked_by(Arg, Ex):-  
    my_fetch(Arg, external_attacked_by, Ex).  
get_directly_supported_by(Arg, Ex):-  
    my_fetch(Arg, external_supported_by, Ex).  
  
get_total_nodes(Arg, Nodecount):-  
    examine_node(Arg,N1),  
    squash(N1,Nodes),  
    length(Nodes,Nodecount1),  
    Nodecount is (Nodecount1/2).  
  
get_weak_nodes(Arg, Weaknodecount):-  
    examine_node(Arg,N1),  
    squash(N1,Nodes),  
    filter(Nodes,Wnodes),  
    length(Wnodes, Weaknodecount1),  
    Weaknodecount is (Weaknodecount1/2).  
  
assess_supported_by(Arg, External_supported_by, V):-  
    assess_root_arg(External_supported_by, V).  
  
assess_attacked_by(Arg, External_attacked_by, V):-  
    assess_root_arg(External_attacked_by, V).
```

```
/* display toulmin structure */
```

```
convert_user_root:-
    previous_user_root(R),
    convert_to_toulmin(R).
```

```
convert_sys_root:-
    previous_sys_root(R),
    convert_to_toulmin(R).
```

```
convert_to_toulmin(R):-
    my_fetch(R, claim, Claim1), pname(Claim1, Claim),
    my_fetch(R, grounds, [Ground1 | _]), pname(Ground1, Ground),
    my_fetch(R, warrant, Warrant1), pname(Warrant1, Warrant),
    current_field(Field),
    gensym(toulminstructure, Win),
    note((the toulminstructure of R is Win)),
    draw_arg_structure(Win, Ground, Claim, ", Warrant, ", Field).
```

```
%% tool
```

```
get_sub_arg(Win, Y, X, Mod):-
    find_pic(Win, (Y,X), Name),
    find_sub_arg(Win, Sub_arg),
    display_sub_arg(Sub_arg).
```

```
find_sub_arg(Win, Support):-
    find_node_with_window(Win, Node),
    my_fetch(Node, supported_by, [Support | _]).
```

```
display_sub_arg([]).
display_sub_arg(Arg):-
    my_fetch(R, claim, Claim1), pname(Claim1, Claim),
    current_field(Field),
    gensym(toulminstructure, Win),
    note((the toulminstructure of R is Win)),
    draw_arg_structure(Win, noground, Claim, ", nowarrant, ", Field).
```

```
find_node_with_window(Win, Node):-
    my_fetch(Node, toulminstructure, Win).
```

```
/* draw toulmin structures */
```

```
create_arg_structure(Window):-
```

```
    wcreate(Window,50,50,350,450,80,200,200,1,1),
    add_tools(Window,[get_sub_arg(icon(0,0,40,40,1001))]),
    gviewer(Window,on).
```

```
draw_arg_structure(Window,Datum,Claim,
```

```
    Modality,Warrant,Backing,Rebuttal,Arg_field):-
```

```
    create_arg_structure(Window),
    draw_box(Window, 'pic1', 10,10),
    concat('<Datum> ',Datum, Datum1),
    draw_text_box(Window, 'btext1',10,10,Datum1),
    draw_box(Window, 'pic2', 10,350),
    concat('<Claim> ',Claim, Claim1),
    draw_text_box(Window, 'btext2',10,350,Claim1),
    draw_box(Window, 'pic2a', -55,225),
    concat('<Modality> ',Modality,Modality1),
    draw_text_box(Window, 'btext2a',-55,225,Modality1),
    draw_box(Window, 'pic3', 100,180),
    concat('<Warrant> ',Warrant, Warrant1),
    draw_text_box(Window, 'btext3',100,180,Warrant1),
    draw_box(Window, 'pic4', 200,180),
    concat('<Backing> ',Backing, Backing1),
    draw_text_box(Window, 'btext4',200,180,Backing1),
    draw_box(Window, 'pic5', 150,350),
    concat('<Rebuttal> ',Rebuttal, Rebuttal1),
    draw_text_box(Window, 'btext5',150,350,Rebuttal1),
    draw_line(Window, line1, 30, 115, 30, 200),
    draw_line(Window, line2, 30, 205, 30, 340),
    draw_line(Window, line3, 35, 202, 95, 202),
    draw_line(Window, line4, 85, 380, 145, 380),
    draw_line(Window, line5, 175, 202, 195, 202),
    draw_line(Window, line6, 25, 280, 18, 280),
    draw_text(Window,text1,25,155,['So']),
    draw_text(Window,text2,65,215,['Since']),
    draw_text(Window,text3,115,390,['Unless']),
    draw_text(Window,text4,185,215,['On Account Of']),
    add_pic(Window, arg_field, grey(double(fillbox(115,10,60,110,70,30)))),
    concat('<<ARGUMENT FIELD>> ',Arg_field,Arg_field1),
    add_pic(Window,arg_field_text,
        textbox('Times',10,1,120,15,50,100,0,Arg_field1)).
```



```

/* convert frames to argnodes */

root_sys_arg(the Att of Obj is Val):-
  my_fetch_facet(Obj, Att, grounds, Grounds),
  my_fetch_facet(Obj, Att, justification, Warrant),
  gensym(sysarg,Arg),
  retractall(current_sys_root(_)),
  assert(current_sys_root(Arg)),
  note((Arg instance_of argument with
    claim: (the Att of Obj is Val),
    grounds:Grounds,
    warrant: Warrant,
    supported_by: [])),
  rest_sys_arg(Arg, Grounds),!.

rest_sys_arg(Arg, []).

% always unsupported here
rest_sys_arg(Arg, [Obj instance_of Class | Rest]):-
  gensym(sysarg,NewArg),
  note((NewArg instance_of argument with
    claim: (Obj instance_of Class),
    grounds:[],
    warrant: none,
    supports: Arg,
    supported_by: [])),
  prove(all supported_by of Arg are OldSupports),
  note((the supported_by of Arg is [NewArg | OldSupports])),
  rest_sys_arg(Arg,Rest).

% supported grounds therefore recur on subgrounds
rest_sys_arg(Arg, [the Att of Obj is Val | Rest]):-
  my_fetch_facet(Obj, Att, grounds, Grounds),
  my_fetch_facet(Obj, Att, justification, Warrant),
  gensym(sysarg,NewArg),
  note((NewArg instance_of argument with
    claim: (the Att of Obj is Val),
    grounds:Grounds,
    warrant: Warrant,
    supports: Arg,
    supported_by: [])),
  prove(all supported_by of Arg are OldSupports),
  note((the supported_by of Arg is [NewArg | OldSupports])),
  rest_sys_arg(NewArg, Grounds),
  rest_sys_arg(Arg,Rest),!.

% unsupported ground
rest_sys_arg(Arg, [the Att of Obj is Val | Rest]):-
  gensym(sysarg,NewArg),
  note((NewArg instance_of argument with
    claim: (the Att of Obj is Val),
    grounds:[],
    warrant: none,

```

```
    supports: Arg,  
    supported_by: [])),  
prove(all supported_by of Arg are OldSupports),  
note((the supported_by of Arg is [NewArg | OldSupports])),  
rest_sys_arg(Arg,Rest).
```

```

/* display argument-1 trees */

udisplay_tree(W,R):-
    create_display_window(W),
    udisplay_itree(W,R,0,0,0,0,140).

%% root
udisplay_itree(W,R,0,0,0,0,Xval):-
    my_fetch(R,claim,Proposition),
    my_fetch(R,warrant,Warrant),
    my_fetch(R,supported_by,Supports),
    display_prop(W,Proposition,0,0),
    Y1 is 80,
    display_prop(W,warrant(Warrant),0,Y1),
    line_it(W,0,0,0,Y1),
    Y2 is 160,
    udisplay_grounds(W,Supports,0,Y1,0,Y2,Xval),!.

%% with supports
udisplay_itree(W,R,OldX, OldY,X,Y,Xval):-
    my_fetch(R,claim,Proposition),
    my_fetch(R,warrant,Warrant),
    my_fetch(R,supported_by,Supports),not Supports = [],
    display_prop(W,Proposition,X,Y),
    line_it(W,OldX,OldY,X,Y),
    Y1 is (Y + 80),
    display_prop(W,warrant(Warrant),X,Y1),
    line_it(W,X,Y,X,Y1),
    Y2 is (Y1 + 80),
    udisplay_grounds(W,Supports,X,Y1,X,Y2,Xval),!.

%% no supports
udisplay_itree(W,R,OldX,OldY,X,Y,Xval):-
    my_fetch(R,claim,Proposition),
    display_prop(W,Proposition,X,Y),line_it(W,OldX,OldY,X,Y),!.

udisplay_grounds(W,[],OldX,OldY,X,Y,Xval). %% this how the list is stored!!
udisplay_grounds(W,[G1 | Rest],OldX, OldY,X,Y,Xval):-
    NewXval is (Xval - 50),
    udisplay_itree(W,G1,OldX,OldY,X,Y,NewXval),
    X1 is (X + Xval), Y1 is (Y + 20),
    udisplay_grounds(W,Rest,OldX,OldY,X1,Y1,Xval),!.

display_prop(W,warrant(Warrant),X,Y):-
    gensym(prop, Pic), gensym(text,Name),
    draw_fancy_box_2(W,Pic, Y,X),
    draw_text_box_2(W,Name,Y,X,Warrant),!.
display_prop(W,the analogue of Obj is Val,X,Y):-
    gensym(prop, Pic), gensym(text,Name), pname(the analogue of Obj is Val,P1),
    draw_fancy_box(W,Pic, Y,X),
    draw_text_box(W,Name,Y,X,P1),!.
display_prop(W,P,X,Y):-

```

```

gensym(prop, Pic), gensym(text,Name), pname(P,P1),
draw_box(W,Pic, Y,X),
draw_text_box(W,Name,Y,X,P1).

draw_box(Window, Pic, Top, Left):-
    add_pic(Window, Pic, box(Top, Left,50,100)).

draw_fancy_box(Window, Pic, Top, Left):-
    add_pic(Window, Pic, double(box(Top, Left,50,100,20,20))).
draw_fancy_box_2(Window, Pic, Top, Left):-
    add_pic(Window, Pic, double(box(Top, Left,50,100,50,50))).

line_it(W,X,Y,X1,Y1):-
    X2 is (X + 50), Y2 is (Y + 50), X3 is (X1 + 50),
    gensym(prop, Line),
    draw_line(W, Line, Y2,X2,Y1,X3).

draw_line(Window, Name,Y1,X1,Y2,X2):-
    add_pic(Window,Name,lines([(Y1,X1),(Y2,X2)])).

draw_text(Window, Name,Y,X,Text):-
    add_pic(Window,Name,text('Times',14,1,Y,X,Text)).

draw_text_box(Window, Name,T,L,Text):-
    T1 is T + 5, L1 is L + 5,
    add_pic(Window,Name,textbox('Times',9,0,T1,L1,40,90,0,Text)).
draw_text_box_2(Window, Name,T,L,Text):-
    T1 is T + 10, L1 is L + 20,
    add_pic(Window,Name,textbox('Times',10,8,T1,L1,30,70,0,Text)).

create_display_window(Window):-
    wgcreate(Window,50,50,350,450,0,200,200,1,1).

```

```
/* code to find weak principle */
```

```
min_valued_principle(B,OutPr):-
    findall([Pr,S],findprinciple(B,Pr,S),Prlist),
    minlist(Prlist, [OutPr,MinS]).
```

```
findprinciple(B,Pr,S):-
    principle(backing_set: B,
              name: Name,
              type: Type,
              claim: Pr,
              grounds : Grounds,
              strength: S).
```

```
maxlist([[K1,Pr]], [K1,Pr]).
maxlist([[K1,Pr],[K2,Pr2] | Rest], Maxset):-
    maxlist([[K2,Pr2] | Rest], MaxRest),
    max1([K1,Pr], MaxRest, Maxset).
```

```
max1([A,B],[C,D],[A,B]):-
    B >= D.
```

```
max1([A,B],[C,D],[C,D]):-
    B < D.
```

```
minlist([[K1,Pr]], [K1,Pr]).
minlist([[K1,Pr],[K2,Pr2] | Rest], Minset):-
    minlist([[K2,Pr2] | Rest], MinRest),
    min1([K1,Pr], MinRest, Minset).
```

```
min1([A,B],[C,D],[A,B]):-
    B <= D.
```

```
min1([A,B],[C,D],[C,D]):-
    B > D.
```

```
principle(backing_set: marxism,
          name: marxist_principle_1,
          type: explanatory,
          claim: cause(class_divisions, conflict_within_groups),
          grounds : common_sense_principle_2,
          strength: 7).
```

```
principle(backing_set: marxism,
          name: marxist_principle_1,
          type: explanatory,
          claim: cause(power, show_of_prestige),
          grounds : common_sense_principle_3,
          strength: 5).
```

```
principle(backing_set: historico_ecological,
          name: h_principle_1,
          type: explanatory,
          claim: cause(invasion, conflict_within_groups),
          grounds : common_sense_principle_1,
```

strength: 8).

backing_for((the social_division of S is class ->
the social_climate of S is conflict_within_groups),marxism).

/* a simple form of Gentner's approach to analogy which depends on analogy rules in the KB */

```

get_artifact_atts(Artefact, Att_list):-
    (Artefact instance_of _ with Body),
    convert_to_list(Body,Att_list).

get_artifact_atts(sys,Artefact, Att_list):-
    sys_belief((Artefact instance_of _ with Body)),
    convert_to_list(Body,Att_list).

get_artifact_atts_s(Artefact, Att_list):-
    (Artefact instance_of _ with Body),
    convert_to_list_s(Body,Att_list).

get_artifact_atts_s(sys,Artefact, Att_list):-
    sys_belief((Artefact instance_of _ with Body)),
    convert_to_list_s(Body,Att_list).

convert_to_list(H:T,[H:T]).
convert_to_list((H,Rest),[H|Restlist]):-
    convert_to_list(Rest,Restlist).

convert_to_list_s(H:T,[H]).
convert_to_list_s((H:T,Rest),[H|Restlist]):-
    convert_to_list_s(Rest,Restlist).

has_attributes(Artefact,Artefact_list):-
    get_artifact_atts(Artefact, Artefact_list).

has_attributes(Artefact,Artefact_list):-
    get_artifact_atts(sys,Artefact, Artefact_list).

has_attributes_s(Artefact,Artefact_list):-
    get_artifact_atts_s(sys,Artefact, Artefact_list).

has_attributes_s(Artefact,Artefact_list):-
    get_artifact_atts_s(Artefact, Artefact_list).

has_second_order_attributes(Artefact, [Att|T]) :-
    get_artifact_atts(Artefact, Artefact_list),
    has_second_order_attributes1(Artefact_list, [Att|T]).

has_second_order_attributes1(Artefact_list, [Att|T]):-
    is_attribute_of_2(Att:[Att1,Att2], Artefact_list),
    is_attribute_of(Att1, Artefact_list),
    is_attribute_of(Att2, Artefact_list),
    remove(Att:[Att1,Att2], Artefact_list,New_A_list), %% others??
    has_second_order_attributes1(New_A_list , T).
has_second_order_attributes1(A, []).

is_attribute_of(Att, Artefact_list):-
    member(Att:_, Artefact_list).

```

```
is_attribute_of_2(Att, Artefact_list):-  
    member(Att, Artefact_list).
```

```
same_atts([Att:_I[]],[Att:_I[]]).  
same_atts([Att:_I Rest1],[Att:_I Rest2]):-  
    same_atts(Rest1, Rest2).  
same_atts_s(L1,L2):-  
    L1 = L2; subset(L1,L2);subset(L2,L1).
```



```
/* sys kb */

assert_sys_kb:-
assert(
sys_belief((
feature13 instance_of structure with
  height: 2,
  width: 2,
  angle: 45,
  use: hide_preparation,
  analogue: unknown))),
assert(
sys_belief((
feature10 instance_of structure with
  height: 2,
  width: 2,
  angle: 45,
  use: cooking,
  analogue: unknown))).

retract_sys_kb:-
retractall(sys_belief(_)).

/* user kb */

assert_user_kb:-
assert(
user_belief((the use of feature11 is pot_making))).

retract_user_kb:-
retractall(user_belief(_))
```

```

/* untested rejigkb */
rejig_kb:-
    overall_assess(Val),
    deal_with_assessment(Val).

deal_with_assessment('neither wins'):-
    nl('ASPA dialogue'),nl('ASPA dialogue'),
    write('ASPA dialogue', ' NEITHER SIDE WINS - no changes to kbs...'),
    wfront('ASPA dialogue').

deal_with_assessment('system wins'):-
    nl('ASPA dialogue'),nl('ASPA dialogue'),
    write('ASPA dialogue', ' ASPA WINS - changing user kb...'),
    wfront('ASPA dialogue').

deal_with_assessment('user wins'):-
    nl('ASPA dialogue'),nl('ASPA dialogue'),
    write('ASPA dialogue', ' USERWINS - changing system kb...'),
    wfront('ASPA dialogue'),
    current_user_root(R),
    my_fetch(R, response_type,Rtype),
    deal_with_rtype(Rtype).

deal_with_rtype(['attack sys claim']):-
    previous_sys_root(R),
    my_fetch(R, warrant, W),
    devalue(W).
deal_with_rtype(['attack sys ground']):-
    previous_sys_root(R0),
    current_user_root(R1),
    actually_attacks(R1,R0,G),
    devalue(G).
deal_with_rtype(['defend user claim']). % no change
deal_with_rtype(['defend user ground']). % no change

actually_attacks(A,Arg,G):-
    ground_of(Arg,G),
    my_fetch(A,claim,C),
    contradicts(C,G).

devalue(Obj):-
    my_fetch(Obj, strength_of_belief,S),
    NewS is (S - 10),
    note(the strength_of_belief of Obj is NewS).

```

Appendix II - The Early Bronze Age 'Wessex' culture

The Early Bronze Age (EBA) 'Wessex' culture of central southern England owes its definition to Piggott's 1938 paper. For the purposes of this appendix the EBA is taken to extend from about 1700BC to 1200BC. The Wessex culture is taken to last from about 1550BC to 1400BC. There has always been controversy as to whether the culture identified by Piggott represents a grouping of real people in the area at the time or whether what he described was an artefact of a particular kind of archaeological analysis. This is not our concern here. I will briefly talk about the culture and then go on to describe the argument I am concerned to simulate: i.e. that surrounding the reasons for the rise of a particular social organization in Wessex at this time.

Piggott defines the classic Wessex culture in terms of the 'presence in grave-groups of certain objects which are either intrinsically early or of peculiarly Wessex types'. These objects are given in the paper as follows:

1. Bronze daggers having midribs and/or lateral grooves;
2. Certain pottery forms, notably the incense cups of the well-known 'Grape Cup' and 'Aldbourne Cup' types;
3. Beads of blue faience, now established as of Egyptian origin;
4. Gold ornaments (of Irish origin);
5. Amber beads and pendants;
6. Stone battle-axes of the Snowhill type;
7. Certain pins of germanic types, and perhaps the flanged axes.

More recently a two-fold division of the Wessex culture (derived from ApSimon) has been accepted. The following (based on Megaw and Simpson, 1979) gives the main characterising features:

Wessex I

- Armorico-British A and B type daggers. These have triangular shaped blades, residual tangs and six rivets. At Bush Barrow in the Wilsford barrow cemetery near Stonehenge a dagger of this type was had a pommel decorated by means of 'thousands of tiny lengths of gold wire... arranged in a chevron pattern'.
- Gold ornaments. These include a gold plate lozenge with geometric design (c. 15cm across), a small lozenge of similar design and a gold 'belt hook' (all from Bush Barrow). Other similar objects are known from Clandon in Dorset and gold button covers, gold bound amber discs and a gold armlet are known from other barrows in the the Wilsford cemetery. Other similar gold objects are known from Little Cressingham in Norfolk. It has been argued that all of this goldwork is the work of a single craftsman or workshop.
- Inhumation burials usually in pits but sometimes on the land surface covered by bowl, bell, disc, saucer or pond barrow types arranged in barrow cemeteries.
- Maceheads are known from Clandon and Bush Barrows.
- Shaft-hole battle-axes are known from Wilsford G58 and Windmill Hill in Wilts.

- Halberd pendants (but not actual halberds) are known from Hengistbury Head in Hampshire, Manton near Marlborough and Wilsford G8.
- Amber jewelry is known from Upton Lovell G2 (the 'Golden Barrow').
- The only pottery known is the so-called 'grape' cups. These are small vessels decorated with round pellets of clay and are known from Manton and Wilsford G58.

Wessex II

- Camerton-Snowhill daggers predominate. These are ogival daggers with thickened midribs and parallel grooves on either side, frequently decorated with a dotted pattern. The hilt is normally attached with three rivets. These are known from Wilsford G23 and Wilsford G56.
- Whetstones perforated for attaching to the belt of the owner are known from Wilsford G23.
- Metal pins with crutch, ring or globe heads.
- Faience beads are common. The commonest form in Wessex is the segmented type but ring, disc and star shaped forms exist.
- The grape cups are replaced by the bipartite vessels known as Aldbourne cups. These vessels have a vertical body, flared neck and are frequently perforated. They are decorated with geometric chevron and lozenge designs.
- Metal hoards are known from Arretton Down, Westbury-on-Trim and Plymstock.
- Cremation is the dominant burial rite with the remains sometimes placed in a collared urn.

Piggot himself is the originator of one point of view in our discussion. While he suggests that the Wessex culture forms an important node in a network of trade stretching from Ireland to Central Europe and the Mediterranean, he stresses that the origins of the culture 'represents an actual immigration' from Brittany. These immigrants brought with them (apart from the daggers) a new burial rite in cremation and the associated bell and disc barrows (often in barrow 'cemeteries'). Since Piggott recognises a parallel burial tradition in the 'cinerary urns' he suggests that the invaders became 'a dominant and intrusive aristocracy who for some centuries at least lorded it over the native element'.

The notion of an invasion from Brittany and hence of an 'intrusive' aristocracy has been rejected by Grahame Clarke (1966). He suggests that both the burial rite and the practice of barrow-building can be traced to earlier local traditions. He also suggests that it is not unlikely that when individuals accumulate wealth and power through the extensive trade (or exchange) networks which must have existed they will show an interest in novel and valuable artefacts.

Grahame Clarke goes on to suggest that it seems to be a 'basic methodological error... to categorize a hundred or so rich burials as though these constituted a culture.' At best the Wessex culture may only represent the burial practices of the better off members of the central southern England region. Concentration on this 'culture' tends to obscure the fact that others in the same area and outside were making do with less

showy burials. At the same time others outside the region were as spendthrift with their wealth. Nonetheless, the rich burials which cluster in the Wessex area represent the most developed and intensive displays of individual wealth and presumably power in the EBA. Thus, while they are not representative of the situation as a whole in the British Isles they do represent a cultural expression which both allows comparison with similar developments in Europe and the Eastern Mediterranean and provides a clear instance of the stratified nature of the society which was beginning to evolve in the EBA.

Our second point of view can be represented as socially oriented and internal. It derives ultimately from a marxist analysis of cultural change. I have not made use of the views of any particular archaeologist but the temper of this view is similar to that held by Clarke, Cowie and Foxon (1985). They suggest that the new material comes about as a result of the leaders of society in Wessex having become part of 'an interregional network of contacts between widespread European groups' and that, further, the particular expressions of the powerful individuals in Wessex were selected and modified from the range of possible symbols available. They also suggest that the power of these individuals came about through their control of the important cult centres at Stonehenge. Thus, they would argue, change comes about as the result of internal tensions rather than external divisions.

Whatever the truth of this particular dispute it offers a good example of two divergent views which are ultimately derived from contrasting models of social change: invasion and internal tensions. For what it's worth, I tend towards the latter view. There are good reasons for thinking that the so-called Wessex culture only represents the activities of a small group of aristocrats. Thus we do not have a whole-scale invasion. Of course we could have a situation analogous to the Norman Invasion where the Anglo-Saxon population was dominated by a relatively small group of militaristic Normans. There are reasons to suppose that the culture derives from indigenous traditions. Thus we are left with the need to explain how this change arose in society. Even if there were no contrary evidence, there are also good reasons for thinking that the particular model adopted by earlier archaeologists (the invasion model) derives as much from the contemporary uncertainties of pre-war Europe as from an objective consideration of the evidence.

It is important to note that as with the hermeneutic circle mentioned in chapter 4 the models are used as a means of interpreting the data. At the same time, to the extent that the data is satisfactorily interpreted (and thus anomalies are explained), the model itself is reinforced and can serve as a more secure grounding for future interpretations of sites of the period.

It is also important to note that the 'anomaly' of the social changes in Wessex depends on a prior interpretation of the changes in the artefacts, burial practices and so on as a social change. This is of course uncertain like any other interpretation. The explanation of this phenomenon might well turn out to be an explanation for a non-existent phenomena.

***Appendix III - Macintosh™ screen snapshots
for example 1(a) in chapter 6***

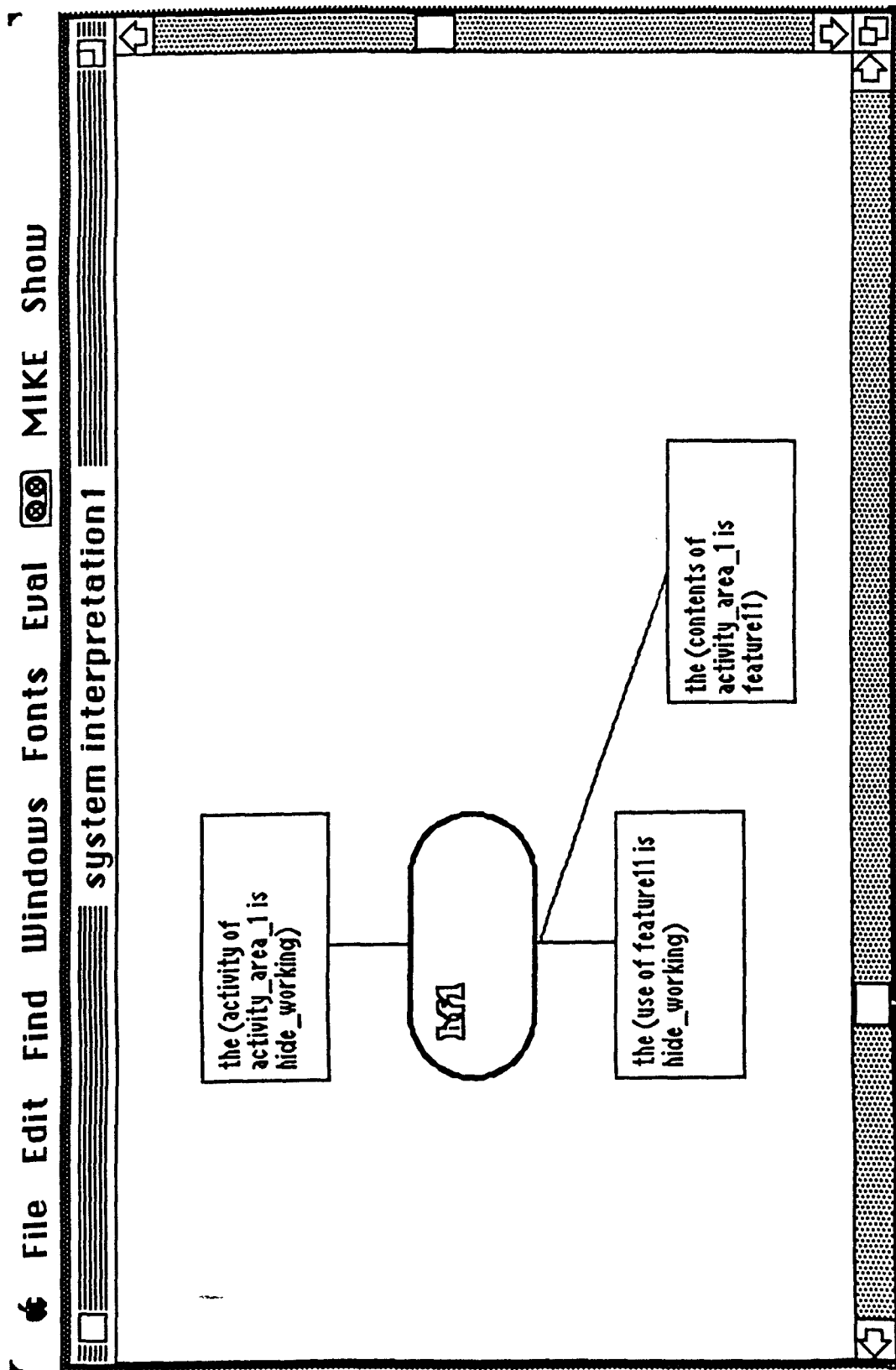


Figure A.1 - Example 1(a) system interpretation 1

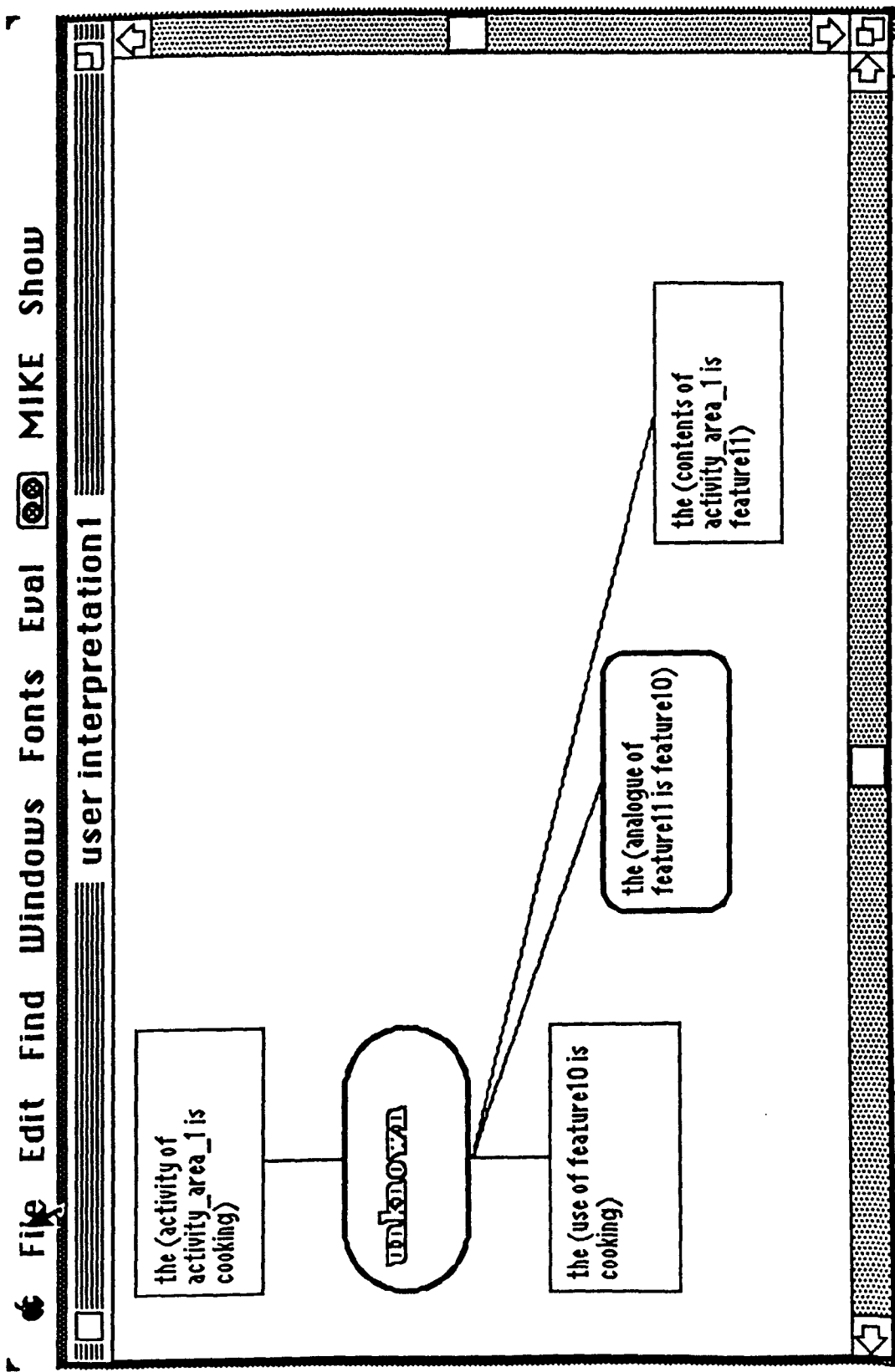


Figure A.2 - Example 1(a) user interpretation 1

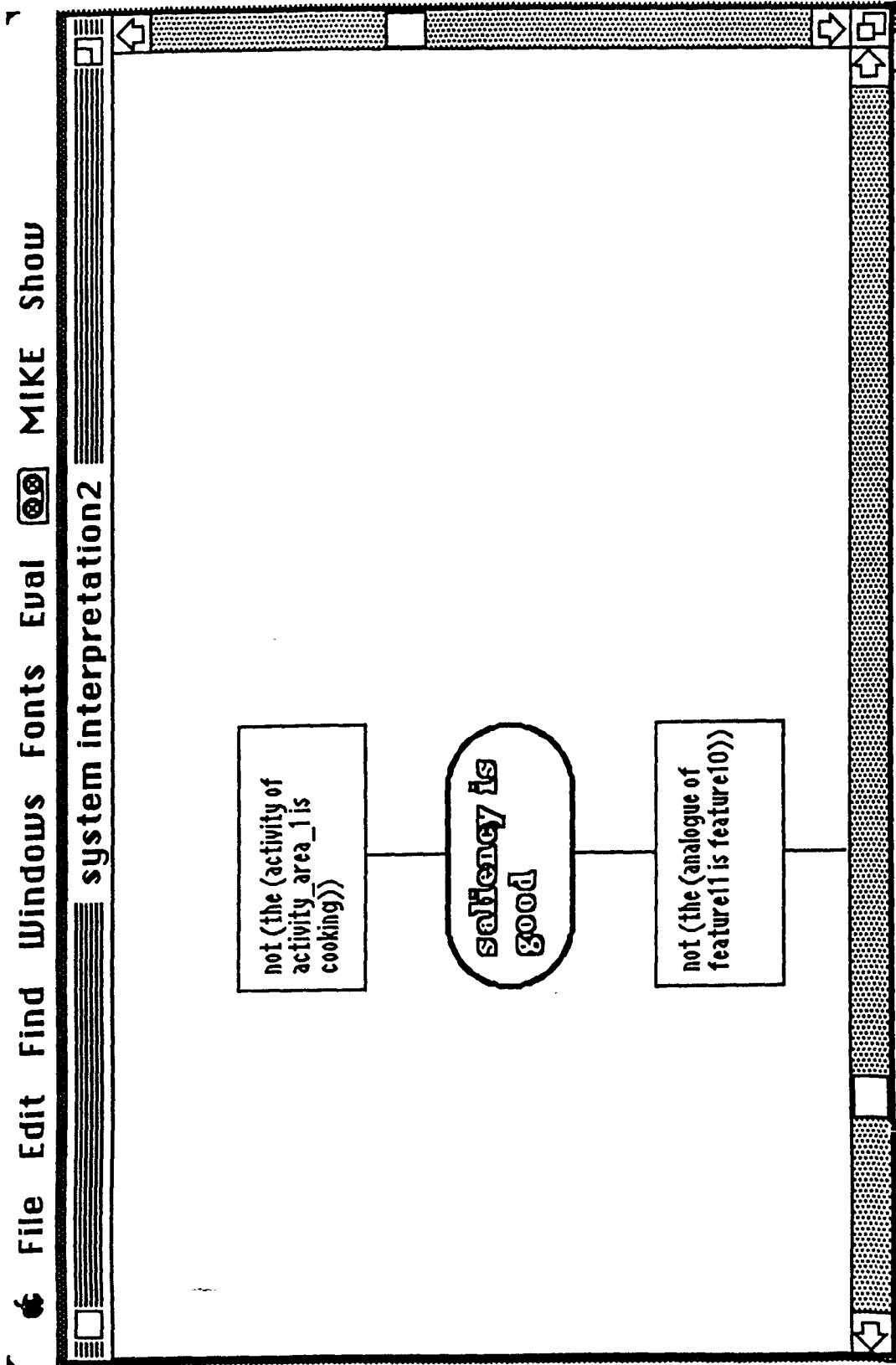


Figure A.3(a) - Example 1(a) system interpretation 2

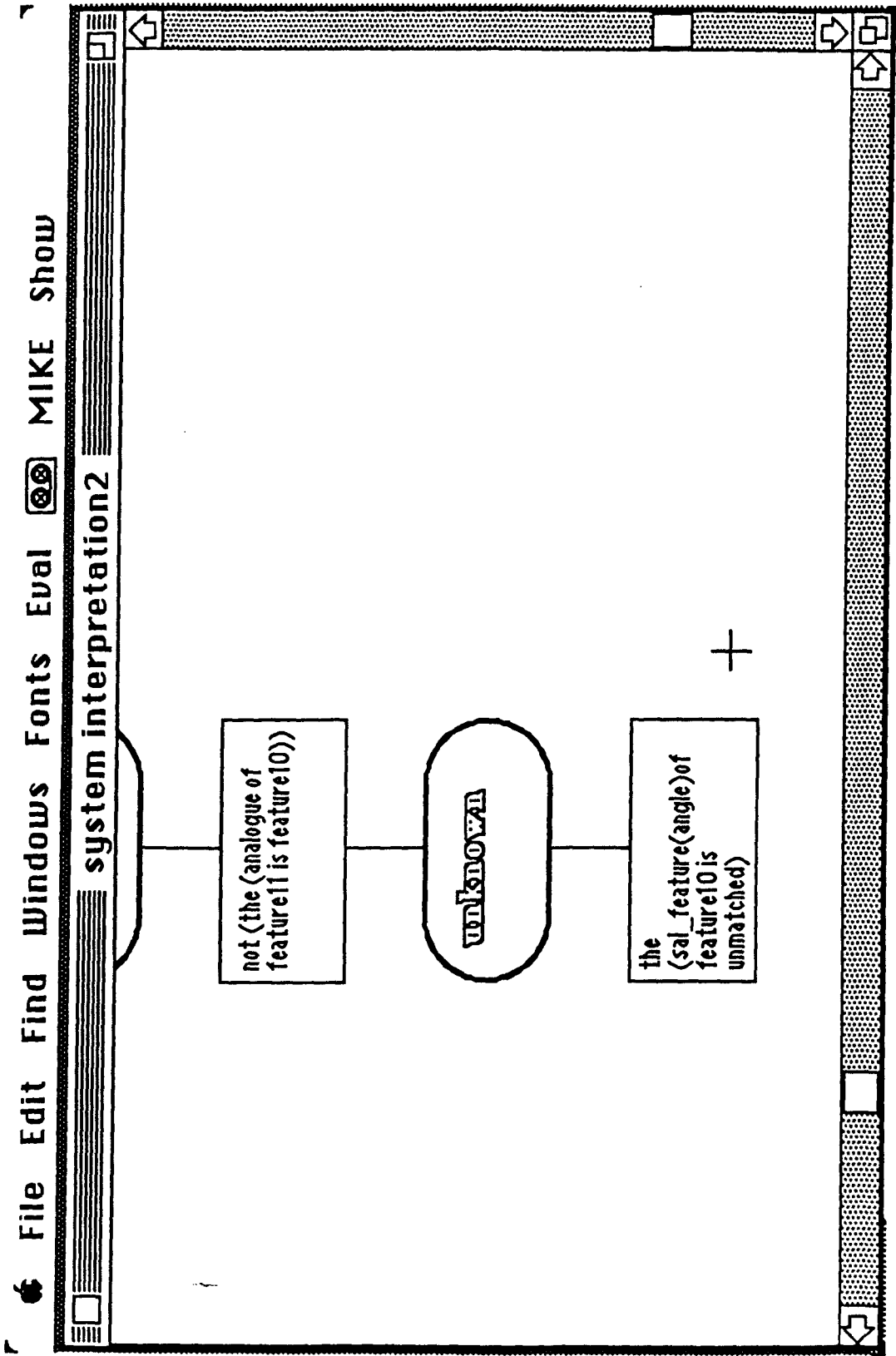


Figure A.3(b) - Example 1(a) system interpretation 2

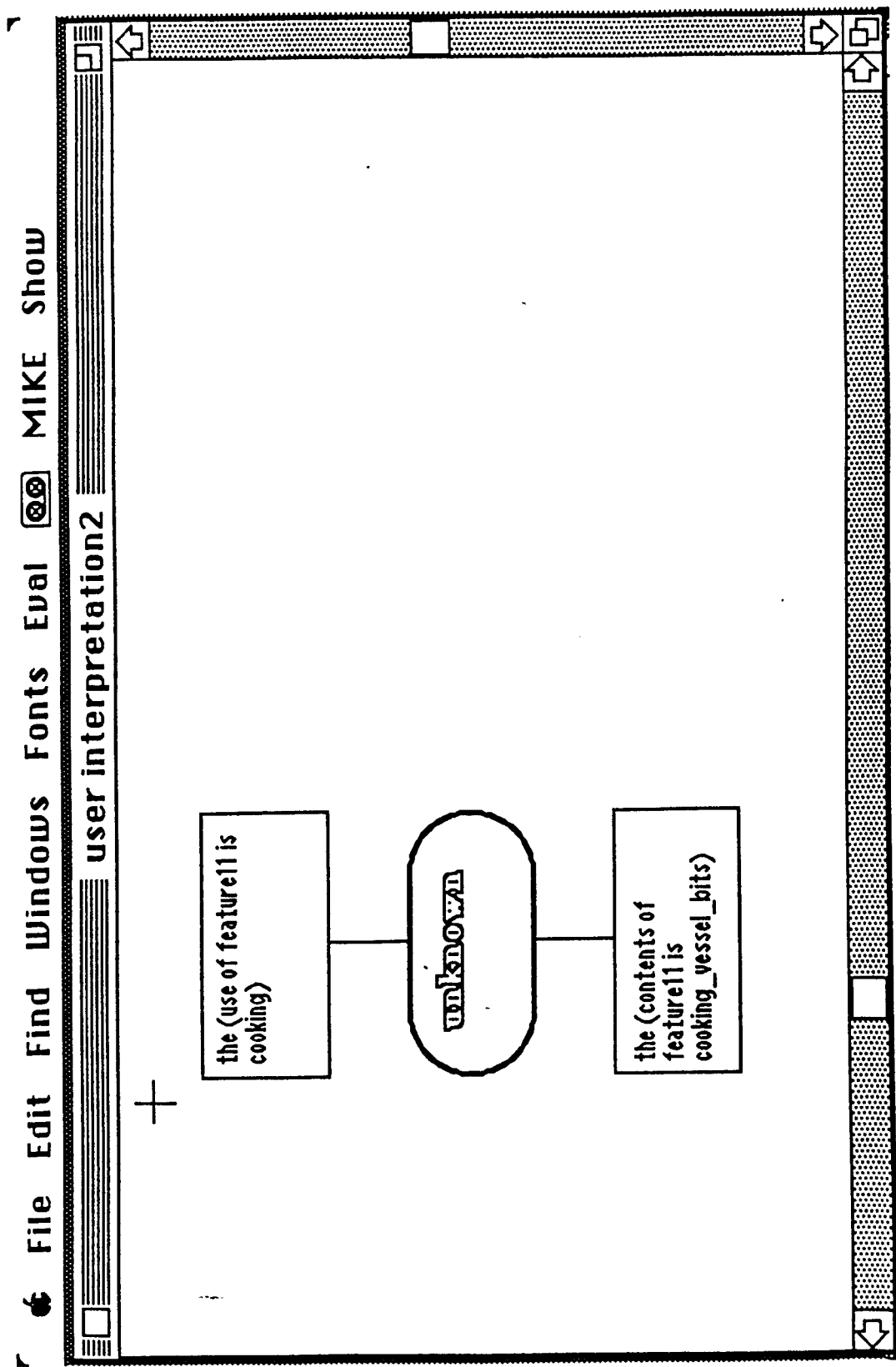


Figure A.4 - Example 1(a) user interpretation 2

*Appendix IV - Results of assessment
algorithm on examples 1-3*

The following table shows the results obtained by running the assessment algorithm discussed in chapter 6 on examples 1-3. I leave it to the reader to gauge the success or otherwise of this algorithm as shown by these results.

example	user	system	winner
1 (a)	7	22	system
1 (b)	7	22	system
2	11	20	system
3	17	20	system

Table A.1 - The results of the assessment algorithm